Propulsion System Effects on the Controllability of a Winged Second-Stage Spaceplane

Msc. thesis

J. Anckaert

Delft University of Technology

TU Delft Delft University of Technology

Challenge the future

PROPULSION SYSTEM EFFECTS ON THE CONTROLLABILITY OF A WINGED SECOND-STAGE SPACEPLANE

MASTER OF SCIENCE THESIS

by

Jorg ANCKAERT

in partial fulfillment of the requirements for the degree of

Master of Science
in Aerospace Engineering

At the Faculty of Aerospace Engineering of Delft University of Technology
2019

Student number:	4304411	
Project supervisor:	Dr.ir. E. Mooij	TUDelft
Thesis committee:	Dr.ir. W. van der Wal	TUDelft,
	Ir. B. T. C. Zandbergen	TUDelft,
	Dr.ir. E. Mooij	TUDelft

Cover image: `https://tinyurl.com/yxkmml8c`

PREFACE

This thesis report describes the work undertaken to obtain the Master of Science degree at the faculty of Aerospace Engineering at the Delft University of Technology. It concerns the hypersonic ascent trajectory of a spaceplane, especially the connection between the propulsion system and stability and control of the vehicle.

First of all, I would like to thank my thesis supervisor, Erwin Mooij, for his guidance and navigation over the course of the past months, and for keeping my more chaotic work habits under control.

Furthermore, I am grateful to my parents for believing in me during my studies and making all of this possible. Additionally, I want to express my gratitude to all my friends, both in Delft and in Belgium, for the continuous support and all the precious moments and ridiculous experiences we have had.

Most of all, I want to thank Birgit for being there for me throughout the good and the bad times, and for all the mental support.

Jorg Anckaert
Delft, July 23, 2019

"It's the questions we can't answer that teach us the most. They teach us how to think. If you give a man an answer, all he gains is a little fact. But give him a question and he'll look for his own answers."

Patrick Rothfuss, *The Wise Man's Fear*

SUMMARY

Spaceplanes are the future of space travel and launching objects into space. Conventional launchers, a.k.a. the common rocket, are becoming obsolete. They are mostly non-reusable and waste lots of resources. The spaceplane is a relatively new concept, with its origin in the 20^{th} century. These vehicles combine the traits of aeroplanes and space vehicles in a clever way to optimise its launching capabilities. They are fully reusable, and provide a lot of flexibility to a mission. Its advantages over current launching systems are equally matched by a series of design and development difficulties. Controlling the attitude of the spaceplane in nominal and off-nominal conditions poses an interesting challenge, certainly when it should only rely on the propulsion system, as is the case for parts of the ascent flight.

Through the investigation of multiple relevant reference concepts, the FESTIP System Study Concept 12 (FSSC-12) upper stage was selected as reference vehicle. This spaceplane stage was derived from another concept, called HORUS. Since this HORUS vehicle was the basis of many other spaceplane concepts, the scope of the study was expanded to the more general HORUS-type spaceplane.

The main research question of this thesis study is defined as follows:

Which parameters and deficiencies in the propulsion system of a winged second-stage spaceplane effect its controllability at single points during the hypersonic ascent phase?

The design points, which are analysed in the simulations, are taken from the FSSC-12 reference vehicle. From this concept, a reference mission is constructed, as well as several mission and system requirements. The nominal conditions are based on the FSSC-12 trajectory and configuration.

To analyse propulsion effects, a dedicated propulsion model is developed to simulate the main rocket engines of the vehicle. The model determines important thrust parameters, which can later be used in the integrated vehicle model. This model includes a propellant tank component, which determines the inertia of the liquid propellants in any flight condition. Sloshing of propellants is an important control problem, and will briefly be discussed as well.

The motion of the vehicle is subjected to multiple forces and moments, of an aerodynamic, propulsive and gravitational nature. These are simulated using a flight simulator: the Generic Attitude and Orbit Control Simulator (GAOCS). This simulator is expanded with the propulsion and inertia models developed for this study. The vehicle is controlled using Nonlinear Dynamic Inversion. This control method provides a straightforward way to control the motion of a vehicle. The control surfaces of the vehicle are not effective at the altitudes of the design points. Therefore, the main rocket engines shall be used to control the vehicle. This method is also called Thrust Vector Control (TVC).

To investigate the controllability of the spaceplane, all the models are integrated and multiple simulations are carried out. The response of the vehicle to step inputs in the aerodynamic angles is analysed for the nominal conditions as a benchmark. Afterwards, off-design cases are investigated, such as engine cut-off, in-flight thrust variations and atmospheric disturbances. Moreover, the thrust sensitivity with respect to the propulsion model input parameters is examined, as well as the sensitivity of the inertia model.

It is demonstrated that the INDI controller can effectively maintain the attitude, as well as perform the step commands. Furthermore, it is established that the thrust sensitivity is within bounds for all test cases, and that the vehicle can still be controlled when one of its three engines fails. Thus, the INDI controller can guarantee the safety of the payload or passengers in many off-design cases, and hence achieve the mission with success. Normal NDI and NDI with TSS were implemented as well, but in a minor degree. There was no conclusive proof for any difference between the three NDI methods.

To conclude, the design of a spaceplane is defined by many interesting problems. Without a dedicated controller, the stability and controllability of the vehicle cannot be guaranteed. However, the TVC-based INDI controller manages to provide a stable and robust method of controlling the vehicle. The safety of the vehicle and its payload can be guaranteed in several off-design and fail cases.

CONTENTS

Preface iii

Summary vii

List of Symbols xiii

List of Abbreviations xvii

1 Introduction 1

2 Mission heritage & System Definition 3
- 2.1 Research Question . . . 5
- 2.2 Vehicle selection . . . 5
- 2.3 FSSC-12 system description . . . 7
  - 2.3.1 Vehicle Configuration and Geometry . . . 7
  - 2.3.2 Aerodynamic Model . . . 8
  - 2.3.3 Propulsion System . . . 9
- 2.4 Reference Mission . . . 9
- 2.5 Mission & System Requirements . . . 10

3 Propulsion Model 15
- 3.1 Chemical Equilibrium Composition . . . 15
  - 3.1.1 Assumptions & Simplifications . . . 15
  - 3.1.2 Governing Equations for Chemical Equilibrium . . . 17
  - 3.1.3 Iteration Equations . . . 17
  - 3.1.4 Thermodynamic Derivatives . . . 17
  - 3.1.5 Iteration Procedure . . . 18
- 3.2 Theoretical Rocket Performance . . . 21
  - 3.2.1 Assumptions & Simplifications . . . 21
  - 3.2.2 Combustion Chamber . . . 22
  - 3.2.3 Throat Section . . . 23
  - 3.2.4 Nozzle Exit . . . 23
- 3.3 Propellant Tank Geometry and Inertia . . . 24
  - 3.3.1 Fill Cases . . . 26
  - 3.3.2 General Approach . . . 27
  - 3.3.3 Case 0: No Inclination . . . 28
  - 3.3.4 Cases A, B, C and D . . . 30
- 3.4 Propellant Sloshing . . . 34

4 Flight Environment 43
- 4.1 Gravity Field . . . 43
- 4.2 Atmosphere . . . 44
- 4.3 Wind Disturbances . . . 45

5 Flight Mechanics **47**
5.1 Reference Frames . . . 47
5.2 State Variables . . . 48
5.2.1 attitude . . . 48
5.2.2 Angular rates . . . 50
5.3 Frame Transformations . . . 50
5.4 Force Models . . . 50
5.4.1 Aerodynamic Moments . . . 50
5.4.2 Propulsion Moments . . . 51
5.5 Equations of Motion . . . 53
5.5.1 Dynamics . . . 53
5.5.2 Kinematics . . . 55
5.6 Vehicle Inertia . . . 55

6 Control System **59**
6.1 Basic NDI Controller for HORUS . . . 59
6.2 NDI Controller with Time Scale Separation . . . 60
6.3 Incremental NDI Controller . . . 61
6.4 Actuator Assignment . . . 62

7 Software Architecture & Verification **65**
7.1 Architecture . . . 65
7.2 Numerical Methods . . . 68
7.2.1 Numerical Integration . . . 68
7.2.2 Interpolation . . . 69
7.3 Verification . . . 70
7.3.1 Chemical Equilibrium Composition & Rocket Performance . . . 70
7.3.2 Tank Inertia . . . 71
7.3.3 Vehicle Inertia . . . 72
7.3.4 Control System . . . 75
7.3.5 Sloshing . . . 77
7.3.6 Other models . . . 77

8 Propulsion & Inertia Model Sensitivities **79**
8.1 Propulsion Model Sensitivities . . . 79
8.1.1 FSSC-12 Upper Stage . . . 81
8.1.2 FSSC-1 . . . 87
8.1.3 Verification Test Cases . . . 88
8.2 Inertia Model Sensitivities . . . 93

9 Simulation Results & Discussion **97**
9.1 Performance Metrics . . . 97
9.2 Nominal System Analysis . . . 99
9.3 Propulsion Effects . . . 103
9.3.1 Thrust Sensitivity . . . 103
9.3.2 In-flight Thrust Variations . . . 105
9.3.3 Propellant Tank Effects . . . 110
9.4 INDI Measurement Noise . . . 116

10 Conclusion & Recommendations **117**
10.1 Conclusion . . . 117
10.2 Recommendations . . . 119

Bibliography **121**

A HORUS-2B Aerodynamic Model **127**

B Frame Transformations **131**

C Gibbs Iteration Equations **133**

D Benchmark Controller **137**
D.1 Feedback Control 137
D.2 Optimal Control Theory 138
D.3 Benchmark: Optimal Feedback Control 139
D.3.1 Longitudinal Control 140
D.3.2 Lateral Control 141
D.4 Benchmark: LQR with Moment Fractions 142

E Linearised Equations & State-Space Representation **145**
E.1 Linearised Equations. 145
E.2 State-Space Representation. 146

List of Symbols

Latin symbols:

$\mathbf{0}$	Zero matrix	-
a	Circle chord	m
a	Radius	m
a	Speed of sound	$\mathrm{m\,s^{-1}}$
a_i, b_i	Polynomial coefficient	-
a_{ij}	Stoichiometric coefficients	-
A	Area / Cross-Section	$\mathrm{m^2}$
$\mathbf{a}$	Scalar quaternion part	-
$\mathbf{A}$	State matrix	various
b	Atomic mass	u
b	Wing span	m
$\mathbf{B}$	Input matrix	various
c	Chord length	m
c^*	Characteristic velocity	$\mathrm{m\,s^{-1}}$
C_p	Specific heat at constant pressure	$\mathrm{J\,kg^{-1}\,K^{-1}}$
C_v	Specific heat at constant volume	$\mathrm{J\,kg^{-1}\,K^{-1}}$
$\mathbf{C}$	Frame transformation matrix	various
$\mathbf{C}$	Output matrix	various
C_D	Drag force coefficient	-
C_l	Roll moment coefficient	-
C_L	Lift force coefficient	-
C_m	Pitch moment coefficient	-
C_n	Yaw moment coefficient	-
C_S	Side force coefficient	-
C_T	Thrust coefficient	-
D	Drag force	N
$\mathbf{D}$	Transmission matrix	various
$\mathbf{F}$	Force vector	N
g	Gravitational acceleration	$\mathrm{m\,s^{-2}}$
G	Universal gravity constant	$\mathrm{m^3\,kg^{-1}\,s^{-1}}$
h	Height/Altitude	
h	Specific enthalpy	$\mathrm{J\,kg^{-1}}$
H	Enthalpy	J
h	Time step	s
$\mathbf{H}$	Mapping matrix	various
I_{sp}	Specific impulse	s
$\mathbf{I}$	Inertia matrix	$\mathrm{kg\,m^2}$
$\mathbf{I}$	Identity matrix	-
J	Cost criterion	various
$\mathbf{K}$	Gain matrix	various

L	Length	m
L	Lift force	N
L^r	rth order Lie derivative	various
$\mathscr{L}$	Aerodynamic roll moment	N m
m	Mass	kg
M	Mach number	-
M	Molecular weight	$\mathrm{kg\,mol^{-1}}$
$\mathscr{M}$	Aerodynamic pitch moment	N m
$\dot{m}$	Mass flow	$\mathrm{kg\,m^{-3}}$
$\mathbf{M}$	Moment vector	N m
n	Moles per unit massof a mixture	$\mathrm{mol\,kg^{-1}}$
$\mathscr{N}$	Aerodynamic yaw moment	N m
NG	Number of gaseous species	-
NS	Number of species	-
OF	Oxygen to fuel ratio	-
p	Pressure	Pa
p	Roll rate	$\mathrm{rad\,s^{-1}}$
P	Pressure ratio	-
$\mathbf{P}$	Covariance matrix	various
q	Pitch rate	$\mathrm{rad\,s^{-1}}$
$\mathbf{q}$	Quaternion	-
$\mathbf{Q}$	State weighting matrix	various
$\bar{q}$	Dynamic pressure	Pa
r	Yaw rate	$\mathrm{rad\,s^{-1}}$
$\mathbf{r}$	Location vector	m
$\mathbf{R}$	Control input weighting matrix	various
R	Atmospheric gas constant	$\mathrm{J\,kg^{-1}\,K^{-1}}$
R	Radius	m
s	Circular arc	m
S	Enthropy	$\mathrm{J\,K^{-1}}$
S	Aerodynamic area	$\mathrm{m^2}$
S	Side force	N
t	Time	s
T	Temperature	°
T	Thrust force	N
u	Velocity	$\mathrm{m\,s^{-1}}$
u	Physical control input	various
U	Gravitational potential	$\mathrm{J\,kg^{-1}}$
$\mathbf{u}$	Input/Control vector	various
v	Number of chemical elements	-
v	Virtual control input	various
V	Velocity	$\mathrm{m\,s^{-1}}$
V	Volume	$\mathrm{m^3}$
w	Width	m
x	Mole fraction	%
x, y, z	Cartesian position coordinates	m
$\mathbf{x}$	State vector	various
X, Y, Z	Axes	-
$\mathbf{y}$	Output vector	various

Greek symbols:

α	Angle of attack	rad
α	Fluid inclination angle	rad
β	Sideslip angle	rad
γ	Specific heat ratio	-
γ_s	Isentropic coefficient	-
γ	Flight-path angle	rad
δ	Latitude	rad
δ_a	Aileron deflection	rad
δ_b	Bodyflap deflection	rad
δ_e	Elevon deflection	rad
δ_r	Rudder deflection	rad
ϵ_T	Thrust elevation angle	rad
ζ	Damping factor	-
η	Efficiency	-
$\eta_{x,y,z}$	Commanded moment fractions	-
θ	Local solar time	s
θ	Outer circle segment angle	rad
θ	Pitch angle	rad
λ	Combustion control variable	-
λ	Lapse rate	$\mathrm{K\,m^{-1}}$
λ	Eigenvalue	various
μ	Chemical potential	$\mathrm{J\,kg^{-1}\,mol}$
μ	Earth's gravitational constant	$\mathrm{m^3\,s^{-2}}$
ξ	Heading angle	rad
ρ	Density	$\mathrm{kg\,m^{-3}}$
σ	Bank angle	rad
τ	Longitude	rad
ϕ	Outer circle segment angle	rad
ϕ	Roll angle	rad
ψ	Yaw angle	rad
Ψ_T	Azimuth angle	rad
ω	Angular rate	$\mathrm{rad\,s^{-1}}$
ω_{cb}	Earth's rotational velocity	$\mathrm{rad\,s^{-1}}$
$\boldsymbol{\omega}$	Angular rate vector	$\mathrm{rad\,s^{-1}}$
$\boldsymbol{\Omega}$	Angular rate skew matrix	$\mathrm{rad\,s^{-1}}$

Sub- and superscripts:

0	Nominal/sea-level condition
$a; aero$	Aerodynamic
a	Ambient
$c; cmd$	Commanded
cc	Combustion chamber
cm	At center of mass
$corr$	Corrected
dry	Empty or dry vehicle state

e	Ellipticity	-
e	Nozzle exit	
G	Gravitational	
i	Index	
j	Species j	
l/L	Left	
L	Limit	
max	Maximum value	
n	Slosh mode	
ref	Reference unit	
r/R	Right	
$tank$	Propellant tank	
th	Nozzle throat	
tot	Total condition	
T	Transpose matrix	
u	Used/Expelled	

Reference frames:

$\mathcal{F}_A$	Aerodynamic reference frame
$\mathcal{F}_B$	Body-fixed reference frame
$\mathcal{F}_P$	Propulsion reference frame
$\mathcal{F}_T$	Trajectory reference frame

Chemical symbols:

CH_4	Methane
CO	Carbon monoxide
CO_2	Carbon dioxide
$COOH$	Carboxyl
H	Hydrogen
H_2	Hydrogen
HCO	Bicarbonate
H_2O	Water
HO_2	Hydroperoxyl
H_2O_2	Hydrogen peroxide
O	Oxygen
O_2	Oxygen
OH	Hydroxide
RP-1	Refined kerosine (jet fuel)

List of Abbreviations

AOCS	Attitude and Orbit Control System
CEA	Chemical Equilibrium with Applications
CFD	Computational Fluid Dynamics
COG	Centre Of Gravity
COM	Centre Of Mass
DOPRI	DOrmand and PRInce
EOM	Equations Of Motion
FAC	Finite Area Combustion
FSSC	Festip System Studies Concept
HORUS	Hypersonic Orbital Reusable Upper Stage
HWM	Horizontal Wind Model
IAC	Infinite Area Combustion
INDI	Incremental Nonlinear Dynamic Inversion
LQR	Linear Quadratic Regulator
LTI	Linear Time Invariant
MATLAB	MATrix LABoratory
MC	Monte Carlo
MF	Moment Fractions
MFLQR	Moment Fractions Linear Quadratic Regulator
MIMO	Multiple-Input Multiple-Output
MMOI	Mass Moment Of Inertia
NA	Not Applicable
NASA	National Aeronautic and Space Administration
NDI	Nonlinear Dynamic Inversion
OMS	Orbital Maintenance System
PD	Proportional Derivative
RCS	Reaction Control System
RF	Reference Frame
RK	Runge-Kutta
RPA	Rocket Propulsion Analysis
SISO	Single-Input Single-Output
TSS	Time Scale Separation
US	United States

1

INTRODUCTION

More than a century has passed since the first flight of an aircraft by the Wright brothers. Since then, the aerospace industry has expanded exponentially, not only to accommodate for the increased tourism, but also to develop a better understanding of the environment, the world and outer space. The development of air- and spacecraft has known many ups and downs throughout history due to reasons such as budget cuts, low technology levels, politics and many more.

Since the beginning of the space race, people have been looking for an economical way to bring payloads into space. This immediately brought forth the need for reusable launch vehicles, such as the spaceplane, which have been studied excessively in the past decades. A spaceplane is a combination of an aircraft and spacecraft in its most general definition. It is a craft, which has the capability to reach space, and return back to Earth through the use of aerodynamic lifting devices. Over time, many different concepts have been proposed and developed, from single stage to orbit spaceplanes taking off and landing horizontally, to multiple stage concepts using combinations between rocket and air-breathing engines. The main driver for the development of spaceplanes is decreasing the cost of bringing payloads and humans into space, by making the launch vehicle reusable and economically viable. Through the use of lifting surfaces and air-breathing engines, spaceplanes can effectively exploit Earth's atmosphere in order to reach higher velocities and eventually orbit. This decreases the amount of required propellant, as opposed to the current launchers for which the atmosphere is just an obstacle. Furthermore, the lifting devices can be used to perform a guided re-entry back into the atmosphere and glide or fly back to a certain predetermined location. It is expected that the demand for spaceplane missions will grow extensively in the years to come. A lot of research on these vehicles must still be done, and the purpose of this thesis is to contribute to this research, by analysing the effects of the propulsion system on the controllability of the spaceplane during the hypersonic ascent phase.

The FESTIP System Studies by Daimler-Benz Aerospace are some of the most thorough studies on spaceplanes, including multiple concepts. For each of the twelve FESTIP concepts, a thorough study was done on their propulsion system, stability and control. Daimler-Benz Aerospace (1996c) gives a summary of the design activities of the FSSC-1 vehicle, including remarks on stability and control, an overview of the propulsion system and several sensitivity studies. Daimler-Benz Aerospace (1996a) does the same for the FSSC-12 concept. The study by MBB (1988) focuses on the HORUS vehicle, especially its guidance and control. This guidance and navigation of HORUS was also investigated by Mooij (1998). Brinkman (2017) demonstrated the performance of knowledge-based control systems during re-entry, and Viavattene (2018) briefly discusses the use of Thrust Vector Control (TVC) during the ascent flight of FESTIP System Studies Concept 1 (FSSC-1), as well as the flying qualities and stability of the vehicle in hypersonic ascent conditions. What seems to be lacking from previous studies is a more detailed investigation in off-design cases concerning the propulsion system and its effect on the motion of the vehicle. Most studies

assume nominal conditions, perfectly functioning actuators and engines, and constant variables. Although Daimler-Benz Aerospace (1996c) discuss the controllability of spaceplanes as well, TVC applications are either omitted or not investigated in depth. It is the purpose of this study to examine these aspects of the vehicle, and in such a way add essential information to the already existing research.

The main research question of this thesis study is defined as follows:

Which parameters and deficiencies in the propulsion system of a winged second-stage spaceplane effect its controllability at single points during the hypersonic ascent phase?

This thesis will focus on the (rocket) propulsion system of a spaceplane, and how it can be used to control the vehicle, as well as how it effects the vehicle's motion in off-design conditions. The propulsion system consists of the rocket engines and propellant tanks. Liquid propellants can move inside the tanks, which changes the inertia of the vehicle. This change in inertia has an impact on the control of the vehicle as well. Furthermore, the propellant can experience sloshing inside the tanks, which creates difficulties for the control system. To control the vehicle, a Nonlinear Dynamic Inversion controller is developed. A controller computes the required control action to eliminate differences in the actual state with respect to the commanded state. Different versions of this NDI controller are implemented in this study to determine which of these provides the most desired response. The three versions are normal NDI (NDI), NDI with time-scale separation (NDI TSS), and Incremental NDI (INDI).

The reference mission, system requirements, and mission requirements are described in Chapter 2. The FESTIP concept FSSC-12 upper stage is chosen as reference, as there is a lot of data available on this vehicle, which is necessary to perform realistic simulations. The propulsion model consists of multiple components: the chemical equilibrium composition, theoretical rocket performance, and propellant tank geometry. These are developed in Chapter 3. The aim of the propulsion model is to have an accurate way to compute the thrust vector and other rocket performance parameters at all design points. Next, the environmental models are given in Chapter 4. Atmospheric disturbances can have a negative effect on the stability of a spaceplane. Especially wind gusts can create a control problem. Flight simulations require equations of motion, reference frames and force models of the external forces. All of these are described in Chapter 5. Chapter 6 elaborates on the controllers. The NDI controllers are developed, as well as a benchmark controller, which is used as a verification tool. When all separate models are developed, they can be integrated into one complete flight simulator. Its architecture is given in Chapter 7, as well as an overview of the used numerical methods and the verification of the models. Finally, the simulator can be used to answer the research questions. Firstly, the propulsion and inertia models are investigated to determine their sensitivities with respect to input variables in Chapter 8. Afterwards, the controllability of the vehicle is examined in Chapter 9. Finally, a concise conclusion on the research, as well as some recommendations for future research, are given in Chapter 10.

2

MISSION HERITAGE & SYSTEM DEFINITION

Spaceplanes have been studied extensively during the past decades. Numerous concepts have been developed, each with their own (dis)advantages. Propulsion and control system development has been a key point in the design of numerous spaceplane concepts, as well as aeroplanes, rockets, satellites, and more mundane vehicles such as cars. There is an incredible amount of knowledge and information on all of these topics. The focus in this study is on all applications of these systems on spaceplanes, or closely related systems.

One of the most detailed and coherent studies on spaceplanes was the FESTIP studies by Daimler-Benz Aerospace. Over a period of several decades, these studies were focused on the development and investigation of multiple spaceplane concepts. For each of the concepts, a thorough study on propulsion, control, flight behaviour, etc. was done to figure out which concepts were the most realistic. Another spaceplane concept that was investigated in depth in the past was the Sänger vehicle (Kuczera et al., 1993), with its HORUS upper stage.

Although propulsion systems were designed in detail for all these concepts, the study of their impact on the stability and controllability of the vehicle were often only approximated or not done in much detail. Many assumptions were usually taken, such as assumptions of fixed COM positions (Daimler-Benz Aerospace, 1996c) or symmetric inertia properties. Furthermore, questions related to deficiencies in the thrust, such as thrust fluctuations, engine-out capabilities, or in general conditions where the propulsion system is not functioning properly, were often left unanswered. It is not very clear yet how different thrust-related parameters impact the flying characteristics of these kinds of vehicles.

Parameters, such as the combustion temperature and pressure, engine dimensions, propellant type, etc., each have a separate influence on the resulting thrust vector. It is important to know how sensitive the vehicle or propulsion system is to changes in the input variables. These questions have been investigated for normal aeroplanes, but since the operational range of spaceplanes is so much larger in terms of conditions it needs to withstand, altitudes it has to reach, and velocities it must travel at, it is of much interest to study these subjects for spaceplanes. Hence, it is also more interesting to focus on the spaceplane trajectory in the hypersonic regime, which usually occurs at higher altitudes. Finally, propulsion can also be used as a control feature of a vehicle. Viavattene (2018) shortly investigated the use of TVC on the FSSC-1 vehicle, but was not able to incorporate thrust deficiencies in this study. From the FESTIP documents, it is clear that TVC is often used on spaceplanes, so investigating this in more detail certainly adds value.

Throughout different spaceplane studies, multiple propulsion methods have been examined. All concepts use rocket propulsion at some point in the mission, since this is still the favoured method for high-altitude, high-velocity propulsion. Vehicles such as the FSSC-1 (Daimler-Benz Aerospace, 1996c) use rocket propulsion for the full trajectory, from lift-off to landing. The same is true for FESTIP concepts 2 to 4. FESTIP concept 5, on the other hand, uses aerospike or high-pressure staged combustion engines, but the result was not very promising from an economic point of view (Dujarric, 1999). After that, the concepts moved from being single-stage vehicles to two-stage vehicle concepts. These mostly use a lower stage with air-breathing propulsion, and an upper stage with rocket propulsion. A good example is FSSC-12 (Daimler-Benz Aerospace, 1996a). Air-breathing propulsion cannot function at high altitudes due to the thin atmosphere. Furthermore, normal jet engines, or even ramjets and scramjets, can only reach limited velocities. In line with the previous argument about focusing on the hypersonic regime, and considering rocket propulsion is the most common propulsion method in this regime at higher altitudes, the focus of this study will also be on rocket propulsion.

Propulsion and inertia effects mainly impact the ascent trajectory of spaceplanes, as re-entry is usually almost unpowered. Furthermore, the propellant tanks are almost empty during re-entry and the landing phase, so inertia is more constant as well. Studies on the re-entry trajectory, including guidance and navigation, have been done by Daimler-Benz Aerospace for FESTIP, but also by for example Brinkman (2017); Mooij (1998); Rijnsdorp (2017) for the HORUS-2B vehicle. Because of these reasons, the ascent trajectory is investigated in this study, with a focus on the hypersonic regime.

A supporting part of the study is to design a suitable controller for the vehicle. Control systems are used in a great number of applications, ranging from aerospace vehicles, to small robotics, to self-driving cars. Hence, the subject has been studies extensively in the past. FESTIP concepts 1 and 9 simulations were done with a linear controller (Daimler-Benz Aerospace, 1996d). The longitudinal (pitch) and lateral (roll) control were split up, and two separate linear controllers were used. Yaw control is not mentioned in this regard. The same was done for concepts 5, 13 and 14 according to Daimler-Benz Aerospace (1996E). Linear controllers are simple of design, but lack in robustness and accuracy. Viavattene (2018) studied the flying qualities and controllability of FESTIP concept 1 using an adaptive control algorithm, and showed this non-linear controller was quite effective. The original controller of the HORUS-2B vehicle was a PID controller (MBB , 1988). Brinkman (2017) developed knowledge-based control algorithms for HORUS-2B, but with a focus on the re-entry flight. A more standard LQR controller has been applied to the same vehicle by Mooij (1998). Linear control definitely has its limits, certainly for studies that involve off-nominal conditions, such as the thrust deficiencies that are investigated in this study. Therefore, a nonlinear controller should be developed.

It is of interest to the community to test different kinds of controllers to investigate what controllers are optimal for spaceplanes or certain conditions. Hence, the decision was made to implement a controller that was not applied to spaceplanes in any of the found literature. The chosen controller is the nonlinear dynamic inversion controller, which is described in Chapter 6. This controller, mainly in its incremental form, was applied by Sieberling et al. (2010) and Wang et al. (2019) to several aerospace vehicles, and proven to have desirable robust performance. From the above considerations, it is possible to extract the research question, as well as several subquestions. These are given in the next section. Afterwards, Section 2.2 describes several reference vehicles.

2.1. RESEARCH QUESTION

The research question, and its subquestions, are stated as follows:

Which parameters and deficiencies in the propulsion system of a winged second-stage spaceplane effect its controllability at single points during the hypersonic ascent phase?

- Which propulsion system parameters have the most effect on the resulting engine thrust?
- What is the sensitivity of the thrust with respect to flight conditions and engine properties?
- How do tank location, size and inclination impact the mass moment of inertia of the spaceplane?
- Can TVC be used to perform manoeuvres at all design points along the trajectory?
- Do the feedback control laws given in Chapter 6 provide a robust method to control the vehicle in case of engine deficiencies?
- Can the vehicle be controlled in off-design cases where part of the propulsion system fails?

2.2. VEHICLE SELECTION

To answer the research questions, it is important to select one or multiple reference vehicles for which data is readily available. The aerodynamic database should be known, as well as the vehicle's trajectory and preferably its mass properties. Otherwise, it is difficult to make a working representation of the spaceplane. Hence, the focus will only be on spaceplanes that satisfy these criteria. Note that it is not necessary to have only one reference vehicle. It is possible to combine the information from several vehicles in one coherent model. The chosen vehicles are the re-entry vehicle HORUS-2B, FESTIP concept 1, Sänger and FESTIP concept 12, the latter two being TSTO concepts. It is interesting to note that some of these are really similar, or even based on one another. For the TSTO vehicles, the focus is on the upper stages, since the focus of this study is mainly on the hypersonic regime of the flight. Stage separation usually happens before extreme velocities are reached, which makes the lower stages less interesting.

HORUS-2B

HORUS-2B (MBB , 1988) is technically not a spaceplane on its own, since it is a second stage to the Ariana-5 launcher. HORUS-2B is fully reusable, as opposed to its first stage. Hence, the full concept does not count as a spaceplane. However, it is included here since the amount of available aerodynamic data on this concept is extensive. Furthermore, several of the other concepts have similarities to this spaceplane, and even used it as their starting point. Therefore, the information on this concept can easily be transmitted to the other concept, without the loss of much accuracy.

HORUS was made to survive the harsh space environment, and can perform a controlled re-entry flight. For this, it has elevons on each wing, which have the combined elevator and aileron functionality to provide pitch and roll control, two rudders for yaw control, and a body-flap for trim (MBB , 1988). The vehicle has a length of 25 m and a wing area of 110 m^2. It can carry a payload of 7 t and has a launch mass of 56 t. Its propulsion system exists of one main engine with a thrust level of 1055 kN in vacuum. Furthermore, it has two de-orbit engines, 14 orbit control thrusters and 18 fine attitude control thrusters. The propellant mass is approximately 30 t. The attitude control of this vehicle during its re-entry flight has been studied extensively by Brinkman (2017) and Mooij (1998).

FSSC-1

The FESTIP System Study Concept 1 (FSSC-1) was the first vehicle concept developed for the Future European Space Transportation Investigations Programme (FESTIP) by Daimler-Benz Aerospace (1996c). The concept is a VTO-HL, SSTO vehicle with delta-shaped wings. The concept was based on HORUS, the second stage of the Sänger vehicle (Kuczera et al., 1993).

The vehicle provides its own thrust for the ascent phase using its eight main rocket engines with a thrust of 1800 kN each. It controls its flight with several aerodynamic surfaces: a body flap, two elevons and two rudders. During ascent, it can use its main engines to control the attitude as well, through the use of thrust vector control (TVC). FSSC-1 has a take-off mass of 900 Mg, which is considerably larger than HORUS' mass. The propulsion system relies on a liquid bi-propellant (LO_2+LH_2), and carries almost 800 t propellants. Apart from the main engines, the vehicle has two OMS thrusters, and twelve RCS thrusters, which use their own dedicated propellants. FSSC-1 has a fuselage length of 59 m, a wing span of 29.36 m, and an aerodynamic reference area of 820 m^2, making it more than double the size of HORUS-2B. Viavattene (2018) studies the stability characteristics and flying qualities of this vehicle during its ascent phase.

SÄNGER

The German Sänger concept is a TSTO HTHL mission. Its development started in the 1960s, but was carried on until the 1990s (Ashford, 2002). The main aim of this mission was to reduce launch costs. Next to this, Europe had striven for independent launch possibilities for a long time. As previously mentioned, Sänger is a two-stage vehicle which consists of an aircraft, the first stage, and an orbiter, the second stage. The second stage is attached to the first one until it reaches about Mach 7, after which they separate.

The first stage is a delta-winged hypersonic aircraft with five turbo-ramjet engines. It has similar control surfaces as HORUS, and provides an initial boost to the upper stage. The orbiter was designed in two versions. One was an unmanned, expandable upper stage, called CARGUS, which could deliver a payload of about 7-8 ton into Low-Earth Orbit (LEO) (Koelle and Kuczera, 1989). The other version was a manned vehicle, able to carry a crew of four people and three ton of cargo. This second version of the orbiter was called HORUS, and is very similar to the HORUS-2B concept.

Both concepts use a LO_2/LH_2 rocket engine to reach orbit. The HORUS second stage has wings to allow for a re-entry and safe flight to the landing location. The vehicle contains two rocket engines which can deliver 700 kN each (Koelle and Kuczera, 1989). The HORUS upper stage has a fuselage length of 32.45 m, a wing span of 17.70 m and an aerodynamic reference area of 338.3 m^2.

FSSC-12

Loosely based on the Sänger concept, the FSSC-12 TSTO HTHL concept was the only air-breathing concept in the FESTIP studies. Even though the concept was derived from Sänger, it was completely redesigned to fit the FESTIP requirements, leading to a completely new configuration for the first stage. The second stage is an enlarged version of the HORUS orbiter (Daimler-Benz Aerospace, 1996b).

The orbiter contains three main engines, having a thrust of 800 kN each. The main engines were already ignited well before the stage separation. From Mach 1.3, the rocket engines of the second stage would provide thrust, using propellants stored in the first stage. This reduces the dry mass of the second stage, but it requires the need for cross-feeding of propellants, as also used in the Space Shuttle. This adds some complexity during stage separation, but the mass savings are worth the additional risks. The orbiter has a fuselage length of 36.9 m, wing span of 19.9 m and aerodynamic reference area equal to 338.3 m^2 (Daimler-Benz Aerospace, 1996a). The lift-off mass is 193 t, which includes 149.45 t propellant.

Among all these reference vehicles, there is one common factor: HORUS. Each of these vehicles is based on this vehicle, and therefore share many properties. Sänger and FSSC-12 are very similar, certainly their upper stages, and FSSC-1 is basically an enlarged version of HORUS. The aerodynamic database of HORUS-2B can therefore be used for these vehicles as well without loosing much accuracy. Viavattene (2018) adapted this database to the FSSC-1 vehicle, since the size differences have an impact on the aerodynamics. The info on Sänger is not as accessible as for the FESTIP studies. In the end, the main reference vehicle for this project is the FSSC-12 upper stage. Whenever there is information lacking for this concept, the other reference vehicles can be used. In Chapter 2.3, more information on this vehicle is given.

In the previous chapter, several interesting reference missions were outlined. Finally, the FSSC-12 concept was chosen as the main reference design. This chapter elaborates on this concept in Section 2.3, where its main specifications are given. It is important to know as much as possible about the reference vehicle, in order to make the models and simulations as accurate as possible. Then, in Section 2.4, an outline of the reference trajectory is stated. This trajectory will form the basis for most of the simulations, as specific trajectory points will be chosen to perform the simulations. Finally, the mission and system requirements are given in Section 2.5.

2.3. FSSC-12 SYSTEM DESCRIPTION

As was stated in the previous chapter, only the FSSC-12 upper stage is studied. Therefore, the information in this chapter will be limited to this vehicle. Daimler-Benz Aerospace (1996a) gives an overview of the complete configuration of both stages. Firstly, the geometrical properties of the vehicle are given in Section 2.3.1. Then, the aerodynamic model is outlined in Section 2.3.2. Finally, the propulsion model is given in Section 2.3.3.

2.3.1. VEHICLE CONFIGURATION AND GEOMETRY

The main dimensions of the FSSC-12 lower and upper stage, as well as some of their mass properties are given in Table 2.1. The first stage is included to give some perspective of the complete mission. Figure 2.1 shows the design of the HORUS-like upper stage.

The figure shows an aerodynamic shape, with the characteristics of a re-entry vehicle. The blunt nose is common among re-entry vehicle, since these shapes have favourable thermodynamic properties. The propellant tanks: two with LO_2, one with LH_2, are visible as well.

Table 2.1: Geometrical and mass data for the FSSC-12 vehicle (Daimler-Benz Aerospace, 1996a)

Parameter	Unit	First stage	Second stage
Total vehicle length	m	75.0	36.9
Fuselage length	m	70.4	36.5
Fuselage width	m	11.7	5.9
Fuselage height	m	3.7	6.1
Wing span	m	33.2	19.9
Aerodynamic reference area	m^2	893.2	338.3
Total LH_2 tank volume	m^3	427.6	244.6
Total LO_2 tank volume	m^3	NA	121.5
Vehicle dry mass	Mg	129.70	30.99
Vehicle Lift-off mass	Mg	357.02	193
Payload mass (Polar orbit)	Mg	193	2.23

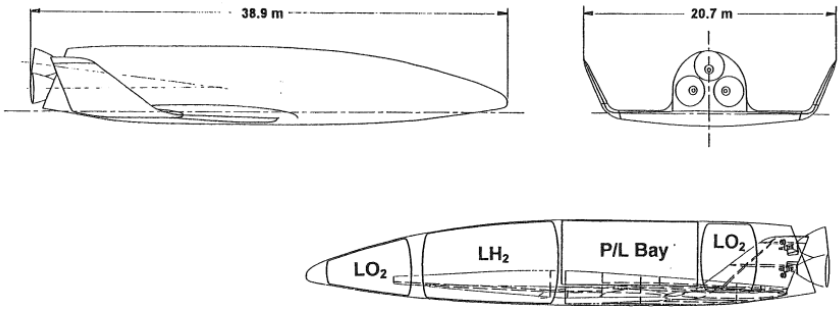


Figure 2.1: Drawings of the FSSC-12 spaceplane upper stage (Daimler-Benz Aerospace, 1996b).

The wings are delta-shaped and have a large surface. This because it needs to create a lot of drag during re-entry. The control surfaces are two rudders, one on each side of the wing, two elevons and a body flap.

The inertia and COM of the vehicle will be computed with a dedicated model. However, the inertia of the empty vehicle should be known. Daimler-Benz Aerospace (1996a) does not give the actual dry inertia, but it lists the inertia properties at landing of the vehicle, which should be relatively close to the empty vehicle.

These parameters are given as follows:

$$\begin{aligned} I_{xx,dry} &= 2.9104 \times 10^5\,\mathrm{kg\,m^2} \\ I_{yy,dry} &= 34.7965 \times 10^5\,\mathrm{kg\,m^2} \\ I_{zz,dry} &= 36.7326 \times 10^5\,\mathrm{kg\,m^2} \\ COM_{x,dry} &= 24.23\,\mathrm{m} \\ COM_{z,dry} &= 1.31\,\mathrm{m} \\ COM_{y,dry} &= 0\,\mathrm{m} \end{aligned} \tag{2.1}$$

Note that the inertia products are assumed to be zero, and the XZ-plane is assumed to be a symmetry plane. More information on the dimensions of the vehicle and its tanks is given by Daimler-Benz Aerospace (1998).

2.3.2. AERODYNAMIC MODEL

To simulate the vehicle performance, it is not only required to establish the dynamic and kinematic EoM. Before the motion can be described in terms of the acting moments and forces on the vehicle, the aerodynamic model must be determined. As the upper stage of the FSSC-12 concept was derived from HORUS, it is assumed the same database can be used. The aerodynamic model of HORUS can hence be derived from the HORUS-2B described by MBB (1988); Mooij (1995). The aerodynamic database of the vehicle is a key point in the simulations. At each time step, the Mach number, altitude and other vehicle states are used to find the corresponding aerodynamic coefficients. These coefficients determine the aerodynamic forces, which in turn influence the vehicle dynamics. The aerodynamic parameters are obtained through interpolation as elaborated on in Section 7.2.2. The model is given in Appendix A.

Table 2.2: FSSC-12 upper stage main engines design specifications (Daimler-Benz Aerospace, 1996a)

Parameter	Unit	Value
Thrust (F)	kN	800
Vacuum specific impulse ($I_{sp,vac}$)	s	465.4
Mass flow rate ($\dot{m}$)	$kg\,s^{-1}$	175.3
Nozzle exit diameter	m	2.380
Maximum combustion chamber pressure ($p_{c,max}$)	bar	244.5
Oxygen to fuel ratio (OF)	-	7.2
Nozzle area ratio	-	275
Number of engines	-	3

2.3.3. PROPULSION SYSTEM

The HORUS winged second stage has its own dedicated propulsion system with 83 Mg propellants. This upper stage must be capable of operating in space and reaching the required orbit, which makes it impossible to use air-breathing engines. HORUS therefore has three rocket engines, with a thrust level of 800 kN each. The vehicle has large propellant tanks, as illustrated in Fig. 2.1. The main engine specifications are given in Table 2.2.

A chamber pressure of 244.5 bar is very large, leading to a high specific impulse. The expansion ratio is large, which is common to most nozzles which have to work efficient at high altitudes. Nozzles with a smaller nozzle ratio are more optimised for sea-level conditions, which is not necessary for HORUS. The specifications given in the table are the nominal design parameters. A dedicated model of the rocket engine is developed in the next chapter. This model will be used to compute the actual values of these variables for different flight conditions.

2.4. REFERENCE MISSION

The reference mission or trajectory is taken from several graphs found in the documents by Daimler-Benz Aerospace (1996a). These graphs show parameters such as altitude, angle of attack and dynamic pressure with respect to the Mach number. This data is not in digital or numeric format, but can be extracted from the graphs using the *engauge digitizer*[1] software. The software return an approximation of the x- and y-values in the graph. The trajectory state variables are given as follows:

$$\mathbf{x}_{traj} = [M, h, \bar{q}, V, \alpha, \beta, \sigma] \tag{2.2}$$

These parameters are required for the fixed-point simulations. From these parameters, others such as the ambient pressure and other atmospheric conditions, can be computed. The graphs by Daimler-Benz Aerospace (1996a) do not include any information about the sideslip and bank angle. However, these angles usually remain constant and equal to zero during the ascent flight, except when certain manoeuvres are carried out. For this study, they will remain zero for the nominal conditions during ascent. This is similar to the trajectory of FSSC-1 as used by Viavattene (2018). Table 2.3 shows the trajectory variables for several points of interest along the ascent flight. For nominal TVC operations, the maximum allowable engine deflections are: $\epsilon_T = \psi_T = \pm 4°$.

[1] `http://markummitchell.github.io/engauge-digitizer/` (Last accessed on May 27, 2019)

Table 2.3: Ascent trajectory of the FSSC-12 orbiter (Daimler-Benz Aerospace, 1996a).

M [-]	**h** [km]	$\bar{\mathbf{q}}$ [Pa]	V [km s^{-1}]	α [°]	β [°]	σ [°]
5	44.9	3269.8	1.690	20.05	0	0
6	52.3	1647.5	2.046	20.00	0	0
7	57.5	1119.1	2.306	18.83	0	0
8	62.2	819.2	2.626	17.73	0	0
9	65.1	659.6	2.843	17.02	0	0
10	67.7	574.8	3.180	16.24	0	0
11	70.1	475.8	3.390	15.36	0	0
12	72.4	388.3	3.634	14.29	0	0
13	74.1	328.0	3.761	13.35	0	0
14	76.2	278.7	4.093	12.27	0	0
15	77.9	250.6	4.388	11.47	0	0
16	79.6	224.6	4.743	10.57	0	0
17	81.2	187.7	4.961	9.76	0	0
18	82.5	152.7	4.928	9.21	0	0
19	84.2	114.9	4.909	8.45	0	0
20	86.4	84.1	4.968	7.58	0	0

2.5. MISSION & SYSTEM REQUIREMENTS

To answer the research questions, it is important to keep the objectives of the study in mind, and to derive several mission and system requirements that will govern the coming work process. The mission requirements are given in Table 2.4, and the system requirements in Table 2.6.

Table 2.4: Mission requirements.

ID	Requirement
MR-1	The analyses shall be done at fixed points in the reference ascent trajectory, with Mach numbers ranging from 5 to 20 as specified in Table 2.3.
MR-2	Basic NDI, NDI with TSS and INDI controllers shall be used to assess the controllability of the vehicle.
MR-3	The propulsion system deficiencies and conditions stated in Table 2.5 shall be included in the research.
MR-4	Nominal conditions for the design points of the FSSC-12 trajectery shall comply with the values stated in the documents by Daimler-Benz Aerospace (1996a).

Table 2.5: Propulsion system deficiencies and conditions.

Topic	Rationale
Constant thrust deviations	How does thrust magnitude influence the response of the vehicle to different commands.
Thrust fluctuations	Thrust fluctuations form a sort of disturbance.Can the control system cope with this?
Engine-out capabilities	Engine failure leads to asymmetric thrust and extra moments. Can these be compensated for?
Engine stuck malfunctions	Stuck engines cannot be used for TVC and induce a disturbance moment. Similar as above.
Changes in tank locations	This is something that cannot happen in-flight, but still interesting to investigate how tank location influences the vehicle response.
Varying empty methods of the propellant tanks	How propellant tanks are depleted has an impact on the mass distribution and inertia. This can lead to asymmetric distributions, which are not favourable for control purposes.

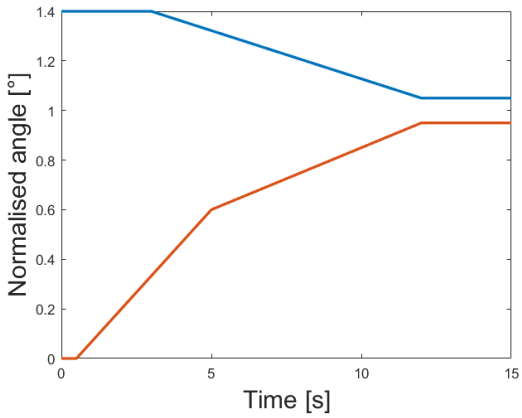


Figure 2.2: Normalised response requirement for step commands in the aerodynamic angles (Daimler-Benz Aerospace, 1996d).

Table 2.6: System requirements - Part 1.

ID	Requirement	Rationale
	Spaceplane system requirements	
SR-SP-1	The FSSC-12 upper stage vehicle shall be used as the main reference vehicle.	See Section 2.2.
SR-SP-2	A consistent and complete aerodynamic database shall be collected before simulations are run.	Without the proper data, results of simulations cannot be interpreted in a correct way.
SR-SP-3	The longitudinal and lateral motion of the vehicles shall be separated for the complete analysis.	
SR-SP-4	The effect of all control surfaces shall be included in the aerodynamic models	
SR-SP-5	Incomplete reference data on FSSC-12 shall be replaced with data from the reference vehicles FSSC-1, Sänger or HORUS-2B.	When data is lacking, similar vehicles can be used to complete the data.
	Propulsion system requirements	
SR-PROP-1	A thrust force in x,y and z direction shall be included in the models for the lower stage.	Addition to most reference documents, where there is no thrust in y-direction.
SR-PROP-2	A dedicated propulsion model shall be developed to compute the thrust related parameters.	Simple equations do not give enough room for deficiency analysis.
SR-PROP-3	The research shall focus on rocket engines	Main method used for most spaceplane reference vehicles at higher altitudes and hypersonic velocities.
SR-PROP-4	The propulsion sensitivity to its input parameters shall be studies.	Objective of the research.
SR-PROP-5	Propellant motion and its effect on the inertia of the vehicle shall be included in the models.	Objective of the research.

Table 2.6: System requirements - Part 2.

ID	Requirement	Rationale
	Control system requirements	
SR-CTRL-1	The control system shall have the ability to provide stable control at all design points of interest.	Objective of the research.
SR-CTRL-2	The normalised response to an angle of attack, bank angle or side-slip angle shall be located within the envelope given in Fig. 2.2.	Requirement for FESTIP system studies Daimler-Benz Aerospace (1996d). The original requirement is only for angle of attack commands, but is extended to the other dimensions in this thesis.
SR-CTRL-3	The controller shall use nonlinear dynamic inversion to control the states.	See Section 2.
SR-CTRL-4	Thrust vector control shall be the main control method during the ascent phase.	Aerodynamic surfaces are not effective at low dynamic pressures.
SR-CTRL-5	The maximum allowable control moment fractions shall be $\eta_x = \eta_y = \eta_z = 1$.	Relates to maximum possible physical control moments, and can not be exceeded.
SR-CTRL-6	The maximum allowable thrust angles are $\epsilon_T = \psi_T = 4°$.	Limitation of the HORUS engines.
	Simulation requirements	
SR-SIM-1	The analysis shall focus only on the attitude or rotational dynamics of the vehicle at the points of interest.	
SR-SIM-2	The general AOCS simulator by Mooij and Wijnands (2002) shall be used as the backbone of the simulations.	This simulator already incorporates most of the equations of motion for a vehicle.

3

PROPULSION MODEL

This chapter describes the theory behind the used propulsion model. The rocket engine model is based on the RPA (Ponomarenko, 2009) and CEA (McBride and Sanford, 1994) software, which are both verified and widely used to analyse the performance of rocket engines.

The model is divided into three main parts: the combustion chamber, throat and exit of the engine. For each of these sections, the equilibrium composition of the reaction products is computed, after which general flow parameters such as temperature, pressure, density, ... can be obtained. Section 3.1 describes the general process of obtaining the chemical equilibrium of the reaction products. After this, the theoretical rocket performance is given in Section 3.2.

Another aspect of the propulsion system is the propellant tanks. Depending on the state of the vehicle, the propellant can move in the tanks, changing the vehicle's inertia. This is described in Section 3.3. Next, the motion of the vehicle can induce sloshing of the propellants. Sloshing is a major issue in terms of control system design, since it creates torques, and changes the state of the vehicle continuously. Furthermore, sloshing forces are sinusoidal with certain natural frequencies. If the control system is not adequately adapted to this, the motion of the vehicle could diverge, and the spaceplane could become uncontrollable. Sloshing is treated briefly in this study, but not in too much detail. The purpose is to make some high-level conclusions about the impact of sloshing. The slosh model is given in Section 3.4.

3.1. CHEMICAL EQUILIBRIUM COMPOSITION

The chemical equilibrium composition routine is the core of the rocket engine model. It determines the amount of each reaction product with respect to the total mixture. The reader is referred to the documentation by Ponomarenko (2009), and McBride and Sanford (1994) for a detailed description of the theory behind this model.

3.1.1. ASSUMPTIONS & SIMPLIFICATIONS

To reduce the complexity of the model, several assumptions were made. These are either to simplify the equations or to reduce the run time of the simulation. The assumptions are as follows:

- **Ideal gases:** For ideal gases, a simplified equation of state can be used:

$$pV = nRT \qquad \frac{p}{\rho} = nRT \tag{3.1}$$

where p is the pressure, V specific volume, T temperature, n the moles per unit mass of mixture, R the universal gas constant and ρ the mixture density.

- **Limited number of reaction products:** As shown by Marchi and Araki (2015), the inclusion of more species at some point does not lead to a large increase in accuracy of the model. Therefore, the number of considered reaction products for each reaction is limited.

- **No condensed species:** During combustion, one or several of the species can change its state from gaseous to condensed, which influences the mixture. From the simulations, it was found that the results without the inclusion of condensed species were already within one percent of the verification results. Therefore, it is not necessary to include these for the studied propellants.

To understand the further sections, some other properties of the mixture need to be discussed. The 'mixture' is defined as the mixtures of the reaction products. It should be distinguished from the mixtures of the reactants, which will be referred to as the 'total reactants'. For example, in the reaction given in Eq. (3.2), H_2 and O_2 are the reactants, and H_2O is the reaction product. The mixture then refers to the reaction products.

$$H_2 + 2O_2 \rightarrow 2H_2O \tag{3.2}$$

Based on the above definitions, n is defined as:

$$n = \sum_{j=1}^{NG} n_j \tag{3.3}$$

where NG is the number of gaseous species, which for this report is equal to the total number of species NS. n_j is the number of kilogram-moles of species j per kilogram of mixture. Then the molecular weight of the mixture is defined as:

$$M = \frac{1}{n} = \frac{\sum_{j=1}^{NS} n_j M_j}{\sum_{j=1}^{NG} n_j} \tag{3.4}$$

with M_j being the molecular weight of each of the species j. Note that in this report, due to the assumption that there are no condensed species, NS and NG are always equal. This not the case in the documentation by McBride and Sanford (1994).

Finally, the mole fractions x_j are defined as:

$$x_j = n_j M_j \tag{3.5}$$

3.1.2. GOVERNING EQUATIONS FOR CHEMICAL EQUILIBRIUM

There are two sets of governing equations based on the type of the problem. The first type is the isobaric-adiabatic problem, further referred to as the hp-problem. It means that the pressure is constant and that there is no heat transfer. The enthalpy H is therefore constant or $dH = 0$. This problem occurs in the combustion chamber of the engine. The second type is the isobaric-isentropic problem. In this case, there is constant pressure and no change in entropy S. This type occurs in the nozzle of the engine during expansion. This type is further referred to as the sp-problem. Both problems are well described in the reports by Ponomarenko (2009) and McBride and Sanford (1994). Therefore, only the resulting governing equations are given below. All relevant symbols can be found in the glossary. The hp-problem consist of a set of $NS + \nu + 2$ equations:

$$\mu_j^0 + RT \ln \frac{n_j}{n} + RT \ln P + \sum_{i=1}^{\nu} \lambda_i a_{ij} = 0 \qquad (j = 1, \cdots, NS) \tag{3.6a}$$

$$\sum_{j=1}^{NS} a_{ij} n_j - b_i^0 = 0 \qquad (i = 1, \cdots, \nu) \tag{3.6b}$$

$$\sum_{j=1}^{NG} n_j - n = 0 \tag{3.6c}$$

$$\sum_{j=1}^{NS} n_j H_j^0 - H_0 = 0 \tag{3.6d}$$

where ν is the number of chemical elements.

The sp-problem is described by the same Eqs. (3.6a)-(3.6c) as the hp-problem, and a different fourth equation given as follows:

$$\sum_{j=1}^{NS} n_j \left(S_j^0 - R \ln \frac{n_j}{n} - R \ln P \right) - S_0 = 0 \tag{3.7}$$

Equation (3.6b) represents the conservation of mass, Eq. (3.6d) the constant enthalpy, and Eq. (3.7) the constant entropy of the system.

3.1.3. ITERATION EQUATIONS

The equations from the previous section are not all linear in the composition variables, which means an iteration procedure is required to obtain the equilibrium composition. As explained by McBride and Sanford (1994), a Newton-Raphson method can be used to solve for the corrections to the initial estimates of the composition variables n, n_j, λ_j and T. The method uses truncated Taylor series. The correction variables are: $\Delta \ln n_j (j = 1, \cdots, NG)$, $\Delta \ln n$, $\pi_i = \lambda_i / RT$, and $\Delta \ln T$. The iteration equations are given in Appendix C.

3.1.4. THERMODYNAMIC DERIVATIVES

To analyse the flow, certain parameters should be obtained from the equilibrium composition. Parameters such as specific heat (C_v or C_p), specific heat ratio ($\gamma = \frac{C_p}{C_v}$), isentropic coefficient (γ_s) and speed of sound (a) are important characteristics of the flow.

The specific heat at constant pressure, C_p, is obtained from Eq. (3.8). Next, the specific heat at constant volume is given in Eq. (3.9) (McBride and Sanford, 1994).

$$\frac{C_p}{R} = \sum_{i=1}^{\nu}\left(\sum_{j=1}^{NG}\frac{a_{ij}n_jH_j^0}{RT}\right)\left(\frac{\partial\pi_i}{\partial\ln T}\right)_p + \left(\sum_{j=1}^{NG}\frac{n_jH_j^0}{RT}\right)\left(\frac{\partial\ln n}{\partial\ln T}\right)_p + \sum_{j=1}^{NS}\frac{n_jC_{p,j}^0}{R} + \sum_{j=1}^{NG}\frac{n_j\left(H_j^0\right)^2}{R^2T^2} \tag{3.8}$$

$$C_v = C_p + \frac{\frac{pV}{T}\left(\frac{\partial\ln V}{\partial\ln T}\right)_p^2}{\left(\frac{\partial\ln V}{\partial\ln p}\right)_T} \tag{3.9}$$

with $\left(\frac{\partial\ln V}{\partial\ln T}\right)_p$ and $\left(\frac{\partial\ln V}{\partial\ln p}\right)_T$ given in Equations (3.10).

$$\left(\frac{\partial\ln V}{\partial\ln T}\right)_p = 1 + \left(\frac{\partial\ln n}{\partial\ln T}\right)_p \quad , \quad \left(\frac{\partial\ln V}{\partial\ln p}\right)_T = -1 + \left(\frac{\partial\ln n}{\partial\ln p}\right)_T \tag{3.10}$$

In the equations above, there are certain derivatives that still need to be determined: $\left(\frac{\partial\pi_i}{\partial\ln T}\right)_p$, $\left(\frac{\partial\ln n}{\partial\ln T}\right)_p$ and $\left(\frac{\partial\ln n}{\partial\ln p}\right)_T$. These are computed using the equations given in Appendix C.

The isentropic coefficient and speed of sound are computed as follows:

$$\gamma_s = -\frac{\gamma}{\left(\frac{\partial\ln V}{\partial\ln p}\right)_T}, \quad a = \sqrt{nRT\gamma_s} \tag{3.11}$$

3.1.5. ITERATION PROCEDURE

The iteration procedure to reach chemical equilibrium is fairly simple. The process above is graphically represented in Fig. 3.1 and goes as follows:

1. **Set input parameters:** The input parameters to the equilibrium composition algorithm are the choice of propellants and fuel/oxidiser mixture ratio, OF. Several parameters related to the chosen propellants are then set, such as the atomic masses, M, of the relevant chemical elements and molecules, the number of chemical elements ν, the considered species, stoichiometric coefficients $\mathbf{a}_{ij}$, assigned number of kilogram-atoms of element i per kilogram of total reactants $\mathbf{b}_0^i$, and the total specific enthalpy h_0 of the reactants. As an example, consider the combustion of liquid hydrogen (H_2) and liquid oxygen (O_2). For these propellants, the considered reaction products are: H_2O, O_2,H_2, OH, O, H, HO_2, H_2O_2. This means that the number of (gaseous) species is $NG = NS = 8$. Atomic masses and molecular weights are readily available from any chemical table. The number of chemical elements is $\nu = 2$ (only O and H). $\mathbf{a}_{ij}$ is given as:

$$\mathbf{a}_{ij} = \begin{bmatrix} a_{O,H_2O} & a_{O,O_2} & a_{O,H_2} & a_{O,OH} & a_{O,O} & a_{O,H} & a_{O,HO_2} & a_{O,H_2O_2} \\ a_{H,H_2O} & a_{H,O_2} & a_{H,H_2} & a_{H,OH} & a_{H,O} & a_{H,H} & a_{H,HO_2} & a_{H,H_2O_2} \end{bmatrix}$$
$$= \begin{bmatrix} 1 & 2 & 0 & 1 & 1 & 0 & 2 & 2 \\ 2 & 0 & 2 & 1 & 0 & 1 & 1 & 2 \end{bmatrix} \tag{3.12}$$

which can be read as: a_{ij} is the number of element i-atoms in the jth reaction species. $\mathbf{b}_i^0$ and h_0 are computed as follows (McBride and Sanford, 1996):

$$\mathbf{b}_i^0 = \frac{\left[OF \frac{a_{O,O_2}}{M_{oxygen}} \quad \frac{a_{H,H_2}}{M_{hydrogen}} \right]}{OF+1}, \quad h_0 = \frac{h_{H_2} + OF h_{O_2}}{OF+1} \tag{3.13}$$

It should be noted that these parameters are different for other fuel/oxidiser combinations. The general procedure for any combination is elaborated on in the documents by McBride and Sanford (1994).

2. **Set initial conditions:** The simulation requires several initial conditions or estimates of the composition variables to run. An initial estimate of T, n and n_j are necessary, as well as one of the pressure. These estimates do not have to be accurate or close to the equilibrium condition, as the algorithm does not have any issue to converge within a few iterations, even with poor initial estimates. As a standard, an initial temperature of T =3800 K, $n = 0.1$ and $n_j = 0.1/NG$ are set. Of course, the result of a previous point can always be used as initial conditions of the new point (e.g. use the throat equilibrium conditions as initial estimates for the exit conditions). The pressure estimate depends on the analysed part of the engine and will be elaborated on in Sections 3.2.2 to 3.2.4.

3. **Compute initial thermodynamic properties:** The equations in this chapter and Appendix C contain several parameters, such a H_j^0, S_j^0, $C_{p,j}^0$ and μ_j^0. These are different for each species, and depend on their temperature. For each species, these thermodynamic parameters can be given in the form of least-squares coefficients (McBride and Sanford, 1994; Ponomarenko, 2009) as follows:

$$\frac{C_p^0}{R} = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4 \tag{3.14a}$$
$$\frac{H^0}{RT} = a_1 + \frac{a_2}{2} T + \frac{a_3}{3} T^2 + \frac{a_4}{4} T^3 + \frac{a_5}{5} T^4 + \frac{b_1}{T} \tag{3.14b}$$
$$\frac{S^0}{R} = a_1 \ln T + a_2 T + \frac{a_3}{2} T^2 + \frac{a_4}{3} T^3 + \frac{a_5}{4} T^4 + b_2 \tag{3.14c}$$
$$\frac{\mu_j^0}{RT} = \frac{H_j^0}{RT} - \frac{S_j^0}{R} \tag{3.14d}$$

with a_i and b_i being the polynomial coefficients, which were taken from the documentation by Burcat and Ruscic (2005) [1]. The above equations and corresponding coefficients can be used for temperatures between 200 K and 6000 K. There is a newer data set which can fit the data to temperatures of 20,000 K, but it will not be used in this study as the temperature will never exceed 6000 K.

4. **Compute A and b matrices:** These matrices are determined according to the equations given in Table C.1.

5. **Solve the system for the correction variables:** The system $\mathbf{Ax} = \mathbf{b}$ can easily be solved in MATLAB for the correction variables: $[\Delta \ln n_j, \Delta \ln n, \Delta \ln T]$. For this simulation, a lower-upper matrix decomposition is used to solve for $\mathbf{x}$ (Ponomarenko, 2009) [2].

6. **Compute convergence parameters:** Occasionally, the iteration equations give extreme corrections, which could lead to divergence if used directly. To compensate for this, a control variable λ is introduced to limit the correction to the state variables at each iteration step. The value of λ depends on the magnitude of the corrections obtained in the previous step. More information on this is given in McBride and Sanford (1994).

7. **Update composition/state variables:** The update step is quite straightforward. The logarithm of the composition variables at step $i+1$ can be computed from the values at step i as follows:

$$\begin{aligned} \ln n_j^{(i+1)} &= \ln n_j^{(i)} + \lambda^{(i)} \left(\Delta \ln n_j\right)^{(i)} \qquad (j = 1, \cdots, NS) \\ \ln n^{(i+1)} &= \ln n^{(i)} + \lambda^{(i)} \left(\Delta \ln n\right)^{(i)} \\ \ln T^{(i+1)} &= \ln T^{(i)} + \lambda^{(i)} \left(\Delta \ln T\right)^{(i)} \end{aligned} \tag{3.15}$$

   The composition variables can then easily be obtained by taking the anti-logarithm from the results of the equations above.

8. **Check convergence criteria:** Several convergence criteria are outlined by McBride and Sanford (1994). The iteration procedure is continued until all relevant criteria are met with, or when the maximum set of iterations has been reached. This last measure is introduced to evade infinite iterations in case a problem occurs.

9. **Finish iteration or repeat steps 2-7:** If convergence is reached, the iteration is terminated. Otherwise, steps 2 to 7 are repeated.

10. **Compute thermodynamic derivatives and flow parameters:** After the iteration, the equilibrium composition is used to compute the flow parameters as given in Section 3.1.4.

[1] `http://garfield.chem.elte.hu/Burcat/burcat.html` (Last accessed on May 27, 2019)
[2] `https://nl.mathworks.com/help/matlab/ref/lu.html` (Last accessed on May 27, 2019)

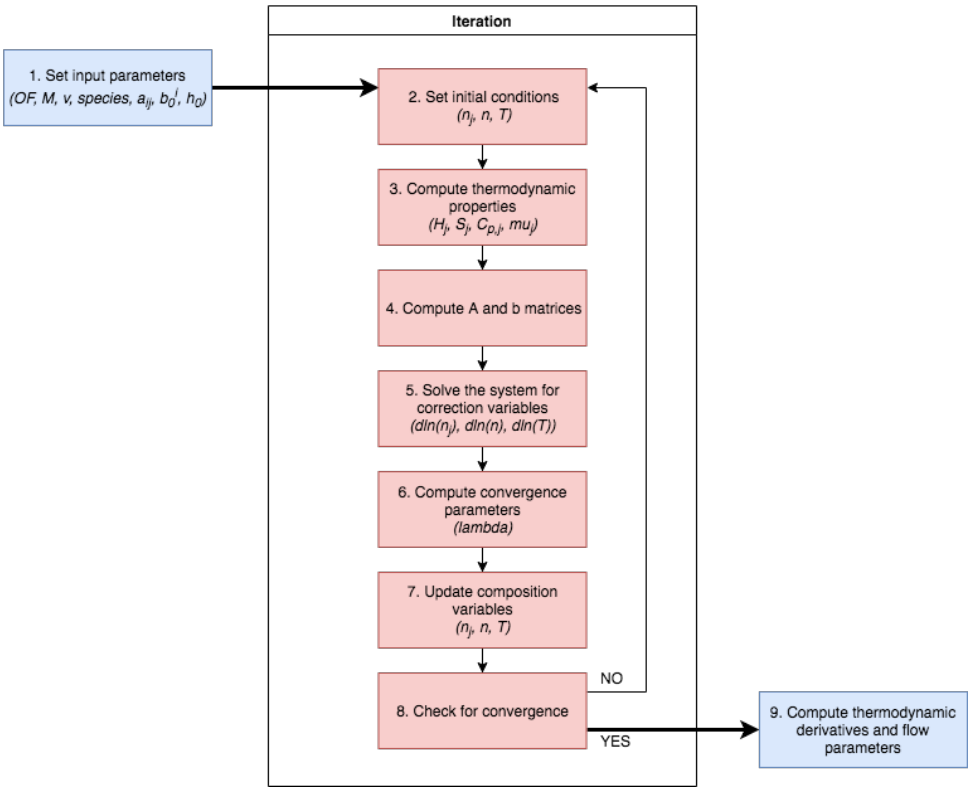


Figure 3.1: Representation of the combustion code iteration process

3.2. THEORETICAL ROCKET PERFORMANCE

This section introduces several rocket performance parameters. The most important of these is the thrust, as this is the main output of the propulsion model. The main rocket performance parameters are given in Table 3.1. The equations in this section are based on the documentation by McBride and Sanford (1994).

Similar to the equilibrium composition algorithm, several assumptions are made to simplify the analysis and equations. These are given in Section 3.2.1. To compute the performance parameters from Table 3.1, the exit flow conditions need to be determined. This is done by first analysing the combustion chamber, then the nozzle throat and finally the nozzle exit. At each of these points, the equilibrium algorithm is used, with the conditions at the previous point as initial conditions. The combustion chamber, nozzle throat and exit are given in Sections 3.2.2 to 3.2.4.

3.2.1. ASSUMPTIONS & SIMPLIFICATIONS

- **Infinite area combustion (IAC):** Assuming an IAC instead of a finite area (FAC) simplifies the calculations, as only one combustion point is required for IAC. FAC requires two points at which the equilibrium composition needs to be computed.

Table 3.1: Rocket performance parameters (McBride and Sanford, 1994).

Parameter	Symbol	Equation	Unit
Thrust	T	$T = \dot{m}u_e + A_e(p_e - p_a)$	N
Specific impulse	I_{sp}	$I_{sp} = \frac{T}{\dot{m}g_0}$	s
Exit Mach number	M_e	$M_e = \frac{u_e}{a_e}$	-
Characteristic velocity	c^*	$c^* = \frac{p_{cc}A_{th}}{\dot{m}}$	$\mathrm{m\,s^{-1}}$
Thrust coefficient	C_T	$C_T = \frac{u_e}{c^*}$	-

- **One-dimensional flow:** The one-dimensional form of the continuity, momentum and energy equations is used, as this greatly simplifies flow equations. These are given in Equations (3.16) respectively. The continuity equation describes the condition of constant mass flow rate: $\dot{m} = \rho A u$.

$$\rho_2 A_2 u_2 = \rho_1 A_1 u_1 \tag{3.16a}$$

$$p_2 + \rho_2 u_2^2 = p_1 + \rho_1 u_1^2 \tag{3.16b}$$

$$h_2 + \frac{u_2^2}{2} = h_1 + \frac{u_1^2}{2} \tag{3.16c}$$

- **Zero velocity at the combustion chamber inlet:** The assumption that the velocity in the combustion chamber is negligible relative to the exit flow velocity, can be introduced in Eq. (3.16c) to compute the exit velocity as given in Eq. (3.17).

- **Isentropic expansion in the nozzle:** Isentropic flow equations are simple, and give straightforward relations between different flow properties. This facilitates the analysis of the nozzle.

- **Supersonic flow after the throat section:** This assumption influences the required equations to compute the exit conditions. For subsonic nozzles, Eq. (3.17) cannot be used.

$$u_e = \sqrt{2\,(h_{cc} - h_e)} \tag{3.17}$$

3.2.2. COMBUSTION CHAMBER

The equilibrium composition in the combustion chamber is an hp-problem, thus with constant pressure and enthalpy. The initial pressure is set to the chamber pressure, which is usually given in the engine specification sheet. The initial enthalpy is obtained from Eq. (3.13).

The combustion temperature is obtained from the equilibrium algorithm. The other relevant parameters, such as discussed in Section 3.1.4 and entropy S can then be computed for the mixture.

3.2.3. THROAT SECTION

As explained previously, the expansion in the nozzle is isentropic, which means entropy is constant. Therefore, the throat, and subsequently nozzle exit, equilibrium composition are sp-problems. The initial entropy value S_0 is equal to the final entropy obtained in the combustion chamber. The pressure in the throat is determined through an iterative process, based on the fact that the Mach number at the throat should be equal to one. Basically, the pressure or pressure ratio for which the Mach number is one needs to be computed.

As an initial estimate, the pressure ratio is approximated as:

$$\frac{p_{cc}}{p_{th}} = \left(\frac{\gamma_s + 1}{2}\right)^{\gamma_s/(\gamma_s - 1)} \tag{3.18}$$

where the combustion chamber value for γ_s is used initially, as γ_s at the throat is not yet known.

This estimated pressure is then used as initial condition for a first equilibrium calculation. From the equilibrium composition, values for u_e and a_e can be determined, which are then used to check the following convergence criterion:

$$\left|\frac{u_e^2 - a_e^2}{u_e^2}\right| \leqslant 0.4 \times 10^{-4} \tag{3.19}$$

If this criterion is not met, the throat pressure is reevaluated using Eq. (3.20), where i indicates the ith iteration. The equilibrium algorithm is applied to every iteration.

$$\frac{p_{th,i+1}}{p_{cc}} = \left(\frac{1 + \gamma_s M^2}{1 + \gamma_s}\right)_{t,i} \tag{3.20}$$

3.2.4. NOZZLE EXIT

Now that the throat conditions are determined, the exit conditions can be obtained. For the following calculations, the area ratio A_e/A_t is required as an input. This is a property of the rocket engine and should be available in the engine documentation. Again, the equilibrium condition is an sp-problem.

Using the assigned area ratio and throat conditions, an estimate of the exit Mach number is obtained as follows (El-Sayed, 2016, p.682):

$$\frac{A_e}{A_{th}} = \frac{1}{M_e}\left(\frac{2 + (\gamma_s - 1)M_e^2}{\gamma_s + 1}\right)^{\frac{\gamma_s + 1}{2(\gamma_s - 1)}} \tag{3.21}$$

after which a pressure estimate is determined:

$$p_e = p_{cc}\left(\frac{1}{\left(1 + \frac{\gamma_s - 1}{2}M_e^2\right)^{\frac{\gamma_s}{\gamma_s - 1}}}\right) \tag{3.22}$$

Another iteration procedure is necessary to find a more accurate value for the exit pressure. First, the equilibrium composition is computed using the above pressure estimate. From the equilibrium composition, the estimated exit velocity can be obtained

with Eq. (3.17). Then, from the continuity equation, an estimated area ratio can be computed:

$$\left(\frac{A_e}{A_{th}}\right)_{est} = \frac{\rho_{th} u_{th}}{\rho_e u_e} \tag{3.23}$$

If the area ratios satisfy the convergence criterion: $\frac{\left|\frac{A_e}{A_{th}} - \left(\frac{A_e}{A_{th}}\right)_{est}\right|}{\frac{A_e}{A_{th}}} \leqslant 0.4 \times 10^{-4}$, the iteration procedure is terminated. If the convergence criterion is not met, the pressure is updated according to Eq. (3.24) and a new equilibrium composition is determined.

$$p_e^{i+1} = p_e^i \left(\frac{(A_e/A_{th})^i_{est}}{A_e/A_{th}}\right)^2 \tag{3.24}$$

After convergence, the exit conditions can be used to calculate all relevant rocket parameters.

Although $\dot{m}$ and A_e are counted as input parameters to compute the thrust, the model makes it impossible to select both beforehand, since these parameters are related through: $\dot{m} = \rho A u$. These parameters do not influence the combustion code, as only the area ratio is used. Only for the thrust equation, these parameters are used. However, the model does compute values for ρ and u at the throat and exit of the nozzle. Together with these values and either $\dot{m}$ or A_e, the other parameter can be computed. Only one of these parameters can be fixed beforehand, whereas the other should be determined at the end. Since it does not really matter which one is chosen as the fixed value, the area is chosen here. This because the dimension of the nozzle is an inherent value of the engine itself. The mass flow is something that can be varied, even during flight, and thus depends more on other parameters. The mass flow can thus be obtained as:

$$\dot{m} = \rho_e A_e u_e = \rho_{th} A_{th} u_{th} \tag{3.25}$$

3.3. PROPELLANT TANK GEOMETRY AND INERTIA

Most tanks in commercial aircraft and spacecraft are either spherical, cylindrical or a capsule shape, since these shapes give the best resistance against the high tank pressures. Spherical tanks are mostly used in spacecraft, whereas the others are more common in aircraft and spaceplanes (Dujarric, 1999). This because spheres do not conform to the general elongated shape of spaceplanes, and lead to more wasted space.

Fitting perfect capsular or cylindrical tanks is usually not straightforward, as the tank geometry needs to conform to the vehicle shape. Figure 2.1 shows that some of the tanks have more complex shapes. However, the main purpose of this study is not to perform an exact finite element method on the structure, but rather show how tank location, inclination angle and mass expulsion influence the stability of the spaceplane. The shape of the tanks is therefore always approximated by a cylinder, which has approximately the same dimensions and volume as the actual tanks. Figure 3.2 shows a horizontal, partially filled, cylindrical propellant tank with liquid height h, length L and radius a.

It is evident that the COM and MMOI of a propellant tank are not constant throughout the ascent phase, as propellant is used continuously. A changing moment of inertia

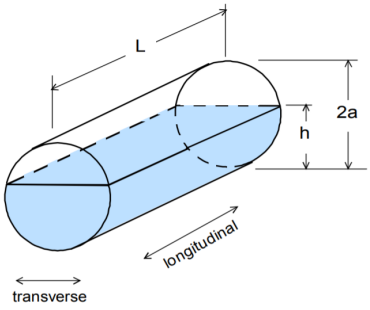


Figure 3.2: Example of a cylindrical, partially filled propellant tank in horizontal position (Dodge, 2000).

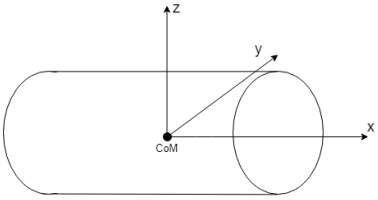


Figure 3.3: The cylinder coordinate system.

does not only have an effect on the motion of the vehicle, but on the control as well. Furthermore, partially filled tanks experience sloshing, which influences the motion of the vehicle, and can potentially be hazardous to the stability. Sloshing of the propellant tanks will be elaborated on in Section 3.4. This section focuses on the inertia and COM change, and how these are modelled in the simulations.

The moment of inertia of a solid cylinder is computed as follows:

$$I_{cylinder} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}mR^2 & 0 & 0 \\ 0 & \frac{1}{12}m\left(3R^2+L^2\right) & 0 \\ 0 & 0 & \frac{1}{12}m\left(3R^2+L^2\right) \end{bmatrix} \tag{3.26}$$

with x, y and z directions as defined by Fig. 3.3. Once propellant is used, the mass distribution in the tanks changes. The MMOI and COM depend on several factors:

- The initial shape of the cylindrical tank.
- The amount of used propellant, or rather the remaining propellant mass or volume that is still present in the tank.
- The inclination angle of the liquid with respect to the horizontal. This angle depends on the acceleration of the vehicle, as the liquid will always be tangent to the acceleration of the vehicle. For example, the fluid is horizontal in a stationary tank, since the gravity is the only force acting upon the fluid, and is directed downwards (vertical).

Table 3.2: Definition of the four fill cases for inclined liquid.

Case	Inclination [°]	Mass	Dimensions	Volume [m^3]
A	$0 \leq \alpha \leq 90$	$0 \leq m_u \leq \frac{1}{2} m_{tot}$	$0 \leq l_A \leq L$ $0 \leq h_A \leq 2R$	Eq. (3.27)[3]
B	$0 \leq \alpha \leq \tan^{-1}\left(\frac{2R}{L}\right)$	$0 \leq m_u \leq m_{tot}$	$0 \leq h_{B,1} \leq 2R$ $0 \leq h_{B,2} \leq 2R$	Eq. (3.28)[4]
C	$\tan^{-1}\left(\frac{2R}{L}\right) \leq \alpha \leq 90$	$0 \leq m_u \leq m_{tot}$	$0 \leq l_{C,1} \leq L$ $0 \leq l_{C,2} \leq 2R$	Eq. (3.29)[5]
D	$0 \leq \alpha \leq 90$	$\frac{1}{2} m_{tot} \leq m_u \leq m_{tot}$	$0 \leq l_D \leq L$ $0 \leq h_D \leq 2R$	Eq. (3.30)[3]

These factors completely determine the location of the COM and the MMOI of the remaining propellant. Based on the parameters above, several cases can be distinguished, each needing a different approach to compute the inertia and COM.

3.3.1. FILL CASES

Fig. 3.4 shows the different fill cases. Some characteristic dimensions and properties of each case are summarised in Table 3.2[3] [4] [5]. As an example, look at case B. Here, the liquid 'cuts' the sides of the cylinder through the circular ends. This case has two characteristic dimensions, which are used in the MATLAB code to define its geometry: the height $h_{B,1}$ and height $h_{B,2}$. This case can occur at any mass level, but not at any inclination. When the inclination gets too steep, this case will change to one of the other cases.

Before the inertia and centre of mass can be calculated, the program should determine which of the fill cases is 'active'. Based on the used mass, tank dimensions, propellant density and inclination angle, a MATLAB algorithm checks which of the fill cases is a possible match for the current situation. For example, when more than half of the mass is expelled, and the inclination angle is larger than the limiting angle of $\alpha_L = \tan^{-1}\left(\frac{2R}{L}\right)$, cases A and B, for which the volume can be computed with Equations (3.27) and (3.28), can immediately be excluded, based on the limits given in Table 3.2. Now the program must decide between the two remaining cases C and D. This is done by solving Equations (3.29) and (3.30) for the characteristic dimensions of each case. These outputs are then compared to the physical limits of the tank to check whether the solution is a true solution. If a solution is found, this case must be the correct one, as there can always only be one case present. This process is verified in Section 7.3.2.

$$V_A = \pi R^2 L - (L - l_A) R^2 \frac{\left[3\sin\phi - 3\phi\cos\phi - \sin^3\phi\right]}{3\left[1 - \cos\phi\right]} \tag{3.27a}$$

$$\phi = \cos^{-1}\left(\frac{h_A}{R}\right) \tag{3.27b}$$

[3] https://rechneronline.de/pi/cylindrical-wedge.php (Last accessed on May 27, 2019)
[4] https://planetcalc.com/1442/ (Last accessed on May 27, 2019)
[5] http://mathworld.wolfram.com/CylindricalSegment.html (Last accessed on May 27, 2019)

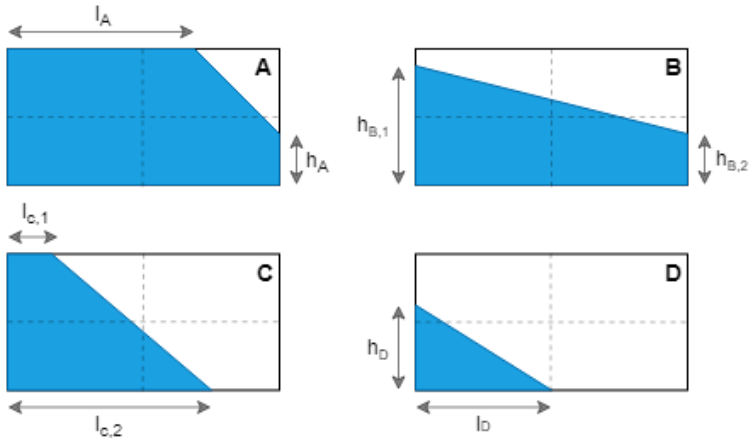


Figure 3.4: Side view of the four fill cases for inclined liquid in a cylinder.

$$V_B = \frac{R^3}{\tan\alpha}\left[K\cos^{-1}K - \frac{1}{3}\sqrt{1-K^2}\left(K^2+2\right) - C\cos^{-1}C + \frac{1}{3}\sqrt{1-C^2}\left(C^2+2\right)\right] \tag{3.28a}$$

$$K = 1 - \tfrac{h_{B,2}}{R}, \quad C = K - \tfrac{L\tan\alpha}{R} \tag{3.28b}$$

$$V_C = \frac{1}{2}\pi R^2 (l_{C,1} + l_{C,2}) \tag{3.29}$$

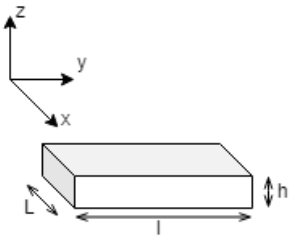
$$V_D = l_D R^2 \frac{\left[3\sin\phi - 3\phi\cos\phi - \sin^3\phi\right]}{3\left[1-\cos\phi\right]} \tag{3.30a}$$

$$\phi = \cos^{-1}\left(\frac{R-h_D}{R}\right) \tag{3.30b}$$

3.3.2. GENERAL APPROACH

Once the fill case is selected, the next stage of the model is initiated where the MMOI and COM are computed. Analytic equations are not readily available for most of the fill cases of a cylindrical tank. Therefore, an approach was thought of, which could be applied to all sorts of possible shapes, without the need for elaborate derivations or integral solving. The MMOI of a point mass with respect to some reference axis equals its mass times the distance squared: $I = mR^2$. This point mass definition can be extended to larger objects, which consist of a sum of point masses.

The MMOI of a multitude of shapes are known exactly, and analytic equations exist. Such shapes are cylinders, spheres, cuboids,... The approach taken here is to divide the body with unknown MMOI in a large number of differential bodies for which the MMOI is known and simple to compute. Division in spheres or cylinders is not really practical.



3

Figure 3.5: Cuboid dimensions and reference frame.

Cuboids, however, are much more suitable to this problem. The MMOI of a cuboid with height h, length L and width l, as shown in Fig. 3.5, is computed as:

$$I_{cuboid} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} = \begin{bmatrix} \frac{1}{12}m\left(l^2+h^2\right) & 0 & 0 \\ 0 & \frac{1}{12}m\left(L^2+h^2\right) & 0 \\ 0 & 0 & \frac{1}{12}m\left(l^2+L^2\right) \end{bmatrix} \tag{3.31}$$

The liquid with unknown MMOI is divided into a large number of cuboids. The MMOI of each of these is computed, and then summed up to get an approximation of the inertia of the liquid. A graphical representation of this is given in Fig. 3.6. Of course, the MATLAB code uses many more cuboids to obtain the MMOI of the liquid, as the accuracy of the method greatly depends on the size of the cuboids. Each case will be treated separately in the following sections.

3.3.3. CASE 0: NO INCLINATION

The case where there is no inclination of the liquid with respect to the horizontal is shown in Fig. 3.6. The liquid level h depends on the tank size, expelled mass and density of the liquid. The total volume of the tank is given as: $V = \pi R^2 L$. The volume of the remaining liquid is obtained from the density and mass: $V_{remaining} = m_{used}/\rho$. Next, the geometry of a circular segment as given in Fig. 3.7 can be used to determine the fill height h. The area in yellow in the figure, equivalent to the area of the liquid in the tank, is obtained from Eq. (3.32). The fill height can be obtained from the inversion of this equation.

$$A = R^2 \cos^{-1}\left(\frac{R-h}{R}\right) - (R-h)\sqrt{2Rh-h^2} \tag{3.32}$$

Once the fill height is known, this value can be divided by the number of required cuboids to find the height of one cuboid. The length of each cuboid equals the length of the tank. Finally, the width l_i of each cuboid needs to be determined, as it depends on the height above the tank base. The width at a certain height h_i is then given by:

$$l_i = 2R\sin(\frac{\theta_i}{2}) \tag{3.33}$$

with θ being the angle as given in Fig. 3.7, which depends on the height h_i.

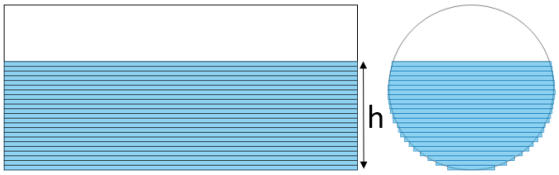


Figure 3.6: Rectangle approximation to compute MMOI of a partially filled, horizontal cylinder (case 0). Side view on the left, and front view on the right.

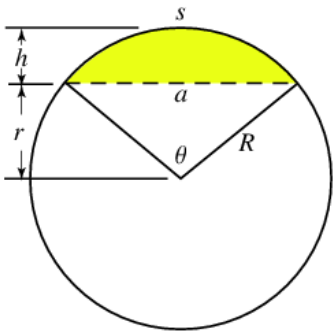


Figure 3.7: Geometry of a circle segment [6].

Figures 3.8 and 3.9 show the results obtained for the partially filled, horizontal cylinder. When the tank is almost empty, all the liquid is located at the bottom of the tank, which results in the low COM_z location. This gradually increases towards the centre of the tank for increasing fill heights. This behaviour is to be expected. Fig. 3.8 shows a jump on the left. This is due to the centroid being located at zero when there is no liquid present, but immediately jumping to about -1 when a small amount of propellant is added to the tank. Note that the mass of the tank itself is not included here, which would remove this sudden jump. If a smaller fill height step would be used to create the graph, the jump would move towards the left.

The MMOI graphs show some interesting behaviour as well. First of all, the MMOI along all directions increases with increasing mass, which is as expected. I_x is always smaller than I_y and I_z, which can be explained by looking at the reference frame. The marks on the graphs show some interesting points, which were used to verify the results. When the cylinder is full, the MMOI is given by Eq. (3.26). The analytic equations for half full cylinders are known as well. Clearly, these points coincide, which verifies the results. Naturally, increasing the dimensions of the tank will increase the inertia properties as well.

[6] `http://mathworld.wolfram.com/CircularSegment.html` (Last accessed on May 27, 2019)

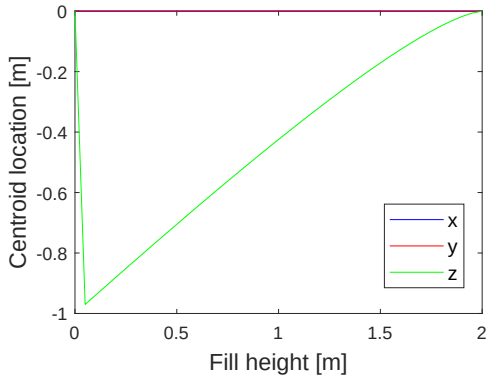


Figure 3.8: COM of a horizontal, partially filled cylinder with L =4 m, R =1 m and ρ =1000 kg m^{-3}.

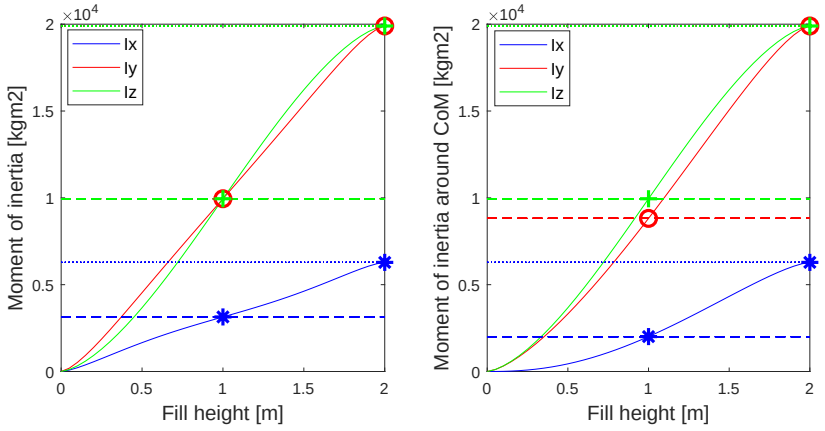


Figure 3.9: MMOI around the centre of the tank around its own COM for a horizontal, partially filled cylinder with L =4 m, R =1 m and ρ =1000 kg m^{-3}. The marks indicate certain verification points (full and half full cylinder).

3.3.4. CASES A, B, C AND D

For case A, only a small amount of propellant (up to half of the total propellant mass in the tank) is used. The angle can range from zero to 90 degrees. The cuboid approximation for this case is shown in Fig. 3.10. Clearly, a different approach needs to be taken here as compared to case 0. The liquid level on both sides of the tank is not equal, and the length of the cuboids, ranging from the complete tank length at the base, towards a smaller value at the top of the tank, is different as well.

Since not all the rectangles have the same length, an expression for these lengths based on the height should be found. The characteristic dimensions for this case are l_A and h_A.

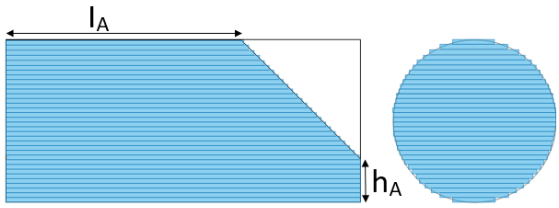


Figure 3.10: Rectangle approximation to compute MMOI of a partially filled, inclined cylinder of case A. Side view on the left, and front view on the right.

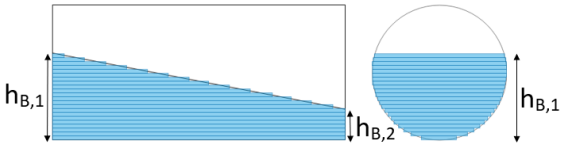


Figure 3.11: Rectangle approximation to compute MMOI of a partially filled, inclined cylinder of case B. Side view on the left, and front view on the right.

The length L_i of a cuboid at height h_i is defined as:

$$L_i = \begin{cases} L & h_i \leq h_A \\ 1 + \frac{2R - h_i}{\tan\alpha} & h_i > h_A \end{cases} \tag{3.34}$$

The width l_i of each cuboid can be obtained with Eq. (3.33).

Case B, shown in Fig. 3.11, is slightly different from the previous case, as the liquid here 'cuts' through both circular sides. The height h_i and width l_i are computed in exactly the same way as for case A. However, there is a slight difference for the length of the cuboids, which is given as follows:

$$L_i = \begin{cases} L & h_i \leq h_{B,2} \\ \frac{h_{B,1} - h_i}{\tan(\alpha)} & h_i > h_{B,2} \end{cases} \tag{3.35}$$

Case C (Fig. 3.12) usually occurs at larger inclination angles. One circular side of the tank is completely covered by the liquid, whereas it does not even touch the other side. Clearly, a liquid distribution like this will lead to a large shift in COM backwards. The height and width are determined as for the previous cases. The length of each cuboid is computed as:

$$L_i = l_{C,2} - \frac{l_{C,2} - l_{C,1}}{2R} h_i \tag{3.36}$$

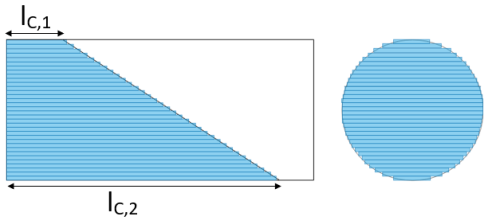


Figure 3.12: Rectangle approximation to compute MMOI of a partially filled, inclined cylinder of case C. Side view on the left, and front view on the right.

Figure 3.13: Rectangle approximation to compute MMOI of a partially filled, inclined cylinder of case D. Side view on the left, and front view on the right.

Case D (Fig. 3.13) occurs at the end of the ascent phase and during re-entry, when most of the propellant has been used. Again, the width and height of each cuboid are determined as for the other cases. The length is determined as follows:

$$L_i = l_D - \frac{l_D}{h_D} h_i \tag{3.37}$$

Now that each case is treated, it is possible to compute the COM and inertia properties of any partially filled inclined cylindrical tank. A simulation was run where the angle was ranged between 0° and 90°, and a full to empty tank. Figure 3.14 shows the results of the fill case selection algorithm. This algorithm basically determines which of the fill cases is active for a certain inclination and remaining propellant mass. This is a quite interesting figure. At low inclinations, case B is the most common. For high inclinations case C is active. This is of course as expected. When the used mass is smaller than half the total mass, the tank will first be in case B for small angles, then experience case A, to finally flow into case D.

The results for COM and MMOI are shown in Figs. 3.16 and 3.15. Overall, the results are as expected. The COM shows large changes, both in x and z direction. At full mass, the COM almost coincides with the centre of the tank, which is clear in the figures as well (e.g. values close to zero). When the angle increases for a certain mass, the COM of the liquid moves back and up. The MMOI increases with increasing mass. As there is always symmetry about the y-axis, the products of inertia I_{xy} and I_{yz} are always zero. It should be noted that the tanks can also have a negative inclination, which translates in the mirror images of the cases shown in Fig. 3.4. The values for COM and MMOI do not change magnitude in this case, but sometimes do have the opposite sign, e.g., for COM_x.

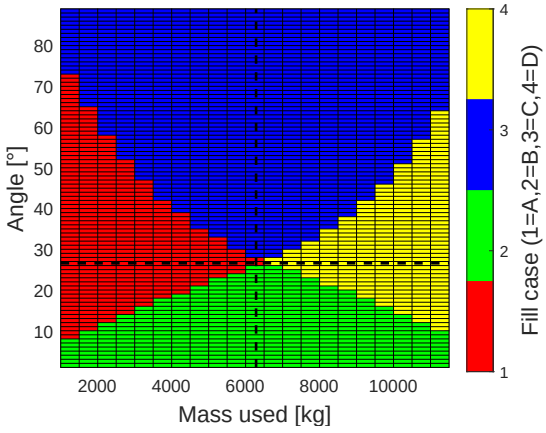


Figure 3.14: Fill case selection for a partially filled, inclined tank for $0 \leq \alpha \leq 90°$ and $0 \leq m_{used} \leq m_{tot}$.

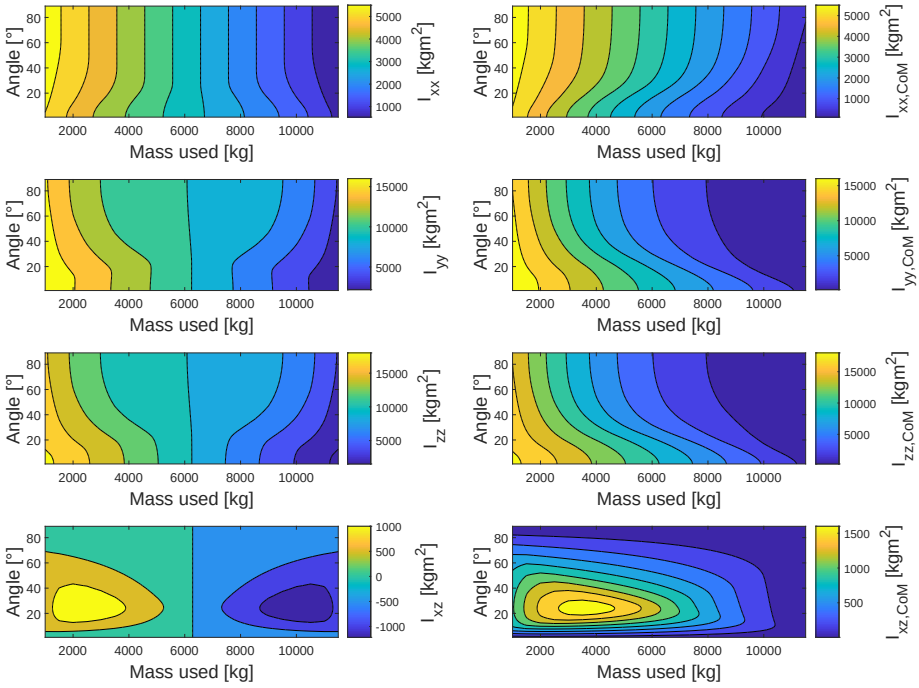


Figure 3.15: MMOI for a partially filled, inclined tank for $0 \leq \alpha \leq 90°$ and $0 \leq m_{used} \leq m_{tot}$. The graphs on the left show the MMOI around the centre of the tank, and the graphs on the right show the MMOI around the COM of the liquid.

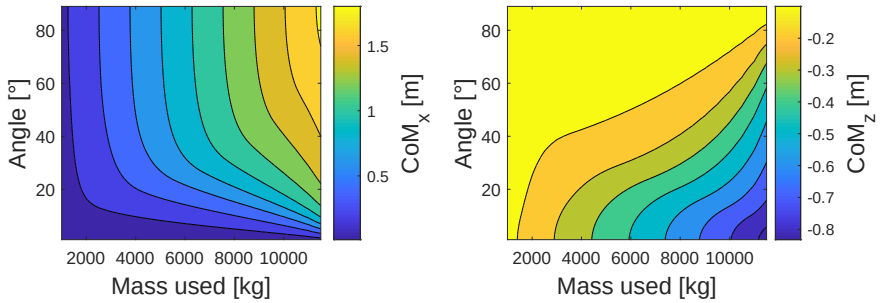


Figure 3.16: COM for a partially filled, inclined tank for $0 \leq \alpha \leq 90°$ and $0 \leq m_{used} \leq m_{tot}$. COM_y is always zero.

3.4. PROPELLANT SLOSHING

Sloshing of propellants can potentially have disastrous consequences on the stability of a vehicle if it is not appropriately accounted for. Newton's laws state that an object or mass in motion or at rest will remain in the same state, unless acted upon by an external force. Secondly, the laws state that when a body exerts a force on another body, it will experience an equal but opposite reaction force from the second body. These principles lie at the origin of the sloshing phenomenon. When the vehicle accelerates as part of a translation or rotation, the liquid in the tanks will not react to this exact motion. The liquid will lag behind or move faster than the surrounding tanks. Due to the limited space in the tanks, the liquid will eventually hit the walls of the tank, and deflect in the opposite reaction. This is called sloshing.

The principles behind the phenomenon are understood very well. Abramson (1966) give an overview of the general sloshing theory in many different tank shapes. The motion of the liquid depends on the tank shape, the liquid properties, and how much liquid is still in the tank. The fuller the tank, the smaller the remaining space in the propellant tank, and hence less room for moving liquids. When the tank is almost empty, the liquids can experience a lot of sloshing motion, but its impact on the vehicle will be small, since the liquid mass is low.

Modelling the sloshing behaviour of liquids is a complex task. For detailed studies, CFD computations are required. The motion of liquids is chaotic at best, and depends on many variables. This kind of study is definitely outside the scope of this project. Another modelling option is approximation of sloshing by an equivalent mechanical model. Essentially, the back-and-forth motion of the liquid is similar to the motion of a spring-mass-damper system or pendulum. The mechanics of these systems are understood very well, and relatively straightforward to model. Raouf (2006) and Basurto et al. (1966) give a detailed overview of these mechanical models and how to apply them to different tank shapes and sloshing cases. This is the approach that will be used in this study. It should be noted that the focus is place on the longitudinal sloshing motion, as opposed to lateral sloshing. This form of sloshing has the most impact on the longitudinal motion, and is coupled with the change in angle of attack, which is the main control variable during ascent.

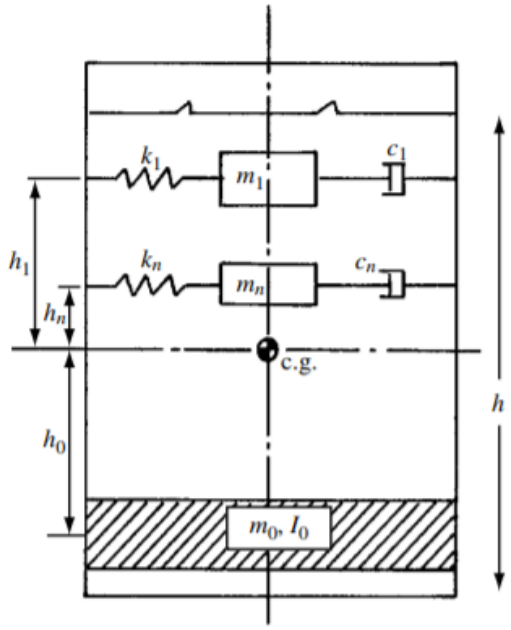


Figure 3.17: Equivalent mechanical spring-mass-dashpot system (Raouf, 2006).

The considered tanks for this study are horizontal cylinders. The sloshing theory for these horizontal cylinders, or also called circular canals, is not as extensive as for multiple other tank shapes. Since this is merely a high-level study on sloshing, and the canal can be compared to a rectangular tank according to Abramson (1966), the properties of the rectangular tank are used whenever these of the horizontal cylinder are not readily available. Abramson (1966) does mention that the rectangular container dimensions should be chosen in such way that the liquid mass of the cylindrical and rectangular tank are the same. This should be done by varying the width of the rectangular tank, and keeping the fill level and length of the tank equal. This is only true for longitudinal sloshing, as a different approach should be taken for lateral motion. Since the liquid volume in the cylindrical tank is known at any point, this can easily be used to determine the width of the equivalent rectangular container through $V = wLh$. Since sloshing is not a major part of this study, and the purpose is mainly to briefly discuss its impact on the model, and how it can affect the motion of the vehicle, this section is only includes the most important parts. The reader is referred to the documentation by Abramson (1966); Basurto et al. (1966); Raouf (2006) for more detailed information on the subject.

The mechanical model is illustrated in Fig. 3.17. The model exists of two components. Firstly, there is a fixed mass m_0 with inertia I_0 at a distance h_0 from the centre of gravity of the tank, which moves in unison with the tank. The sloshing motion consists of different modes. Then, the modes are approximated by a series of spring-mass-dashpot systems.

Each has a mass m_n, spring constant k_n, dashpot c_n, and distance h_n from the COG, where n is the number of slosh modes. For the mechanical model to be equivalent to the moving liquid, the model should satisfy the following conditions (Raouf, 2006):

- *Conservation of mass:* The sum of the fixed mass and sloshing masses should equal the total liquid mass:

$$m_F = m_0 + \sum_{n=1}^{\infty} m_n \tag{3.38}$$

- *Equivalent inertia:* The sum of the inertia terms should equal the inertia of the liquid:

$$I_F = I_0 + m_0 h_0^2 + \sum_{n=1}^{\infty} m_n h_n^2 \tag{3.39}$$

- *Conservation of centre of mass:* The COM of the equivalent model must be equal to the original case:

$$m_0 h_0 + \sum_{n=1}^{\infty} m_n h_n = 0 \tag{3.40}$$

The dashpot can be related to the damping factor: $c_n = 2 m_n \omega_n \zeta_n$, with ζ_n the damping factor of the nth slosh mode given as:

$$\zeta = C \left(\frac{\nu}{d^{3/2} \sqrt{g}} \right)^n \tag{3.41}$$

with d the characteristic dimension of the tank. The constant C equals 1, and the exponent n is 0.5 for rectangular containers, according to Raouf (2006). However, this relation is only true for $h/w \leq 1$, hence when the fill level is higher than the width of the tank. ν is the kinematic viscosity of the liquid. For the sloshing masses, the spring constant, slosh mass and natural frequency ω_n of the sloshing mode are related as:

$$\omega_n^2 = \frac{k_n}{m_n} \tag{3.42}$$

The expressions for the slosh parameters depend on the shape of the tank. Since the circular canal can be approximated by a rectangular tank, the non-dimensional slosh parameters are given as follows (Raouf, 2006):

$$\frac{m_n}{m_F} = \frac{8}{\pi^3} \frac{\tanh\left((2n-1)\pi h/l\right)}{(2n-1)^3 h/l} \tag{3.43a}$$

$$\frac{h_n}{h} = \frac{1}{2} - \frac{\tanh\left((2n-1)\pi h/2l\right)}{(2n-1)\pi h/2l} \tag{3.43b}$$

$$\frac{m_0}{m_F} = 1 - \sum_{n=1}^{\infty} \frac{m_n}{m_F} \tag{3.43c}$$

$$\frac{h_0}{h} = -\frac{1}{(m_0/m_F)} \sum_{n=1}^{\infty} \frac{m_n}{m_F} \frac{h_n}{h} \tag{3.43d}$$

$$\frac{hk_n}{m_F g} = \frac{8\tanh^2\left((2n-1)\pi h/l\right)}{(2n-1\pi)^2} \tag{3.43e}$$

$$\frac{I_0}{m_F h^2} = \frac{I_F}{m_F h^2} \left\{ 1 - \frac{4}{1+(h/l)^2} + \frac{768}{\pi^5 (h/l)\left(1+(h/l)^2\right)} \sum_{n=1}^{\infty} \frac{\tanh\left((2n-1)\pi h/2l\right)}{(2n-1)^5} \right\} - \frac{m_0}{m_F} \left(\frac{h_0}{h}\right)^2 - \sum_{n=1}^{\infty} \frac{m_n}{m_F} \left(\frac{h_n}{h}\right)^2 \tag{3.43f}$$

with h the height of the liquid in the tank and l the length of the tank, and g the gravitational acceleration. The tank aspect ratio is given as h/l.

The results for these parameters are presented in Fig. 3.18. Looking at the mass ratios (top left), there is a clear change of the mass distribution with respect to increasing aspect ratio. The lower the fluid level in a specific tank, the higher the proportion of the mass that is sloshing. For low fluid levels, most of the liquid experiences sloshing. Similarly, for a fixed liquid level, the proportion of sloshing mass increases with the length of the tank. Furthermore, it is evident that the first sloshing mass is definitely the largest. For most applications, it is possible to neglect all sloshing modes but the first (Mooij and Gransden, 2019).

The upper right figure shows the limitations of the model. For aspect ratios smaller than 0.5, the h_0/h ratio seems to diverge. The model does not work properly at this point, so these conditions should be avoided during simulation. The mechanical model can only approximate the reality to a certain point.

The spring constant remains relatively constant for aspect ratios larger than 0.5. This indicates that the natural frequency squared will scale with the sloshing masses according to Eq. (3.42). Hence, larger sloshing masses will have a lower natural frequency. This means the first mode, which contains most of the sloshing masses, will move relatively slow, whereas the higher modes have smaller masses, but have a higher periodicity.

To have a better idea how these variables vary for an actual tank, an example of one of the FSSC-12 LO_2 tanks is given below. LO_2 is chosen since this liquid has a much higher density than LH_2, and will therefore have a larger impact on the vehicle motion. The tank parameters are given in Table 3.3, and the slosh parameters in Fig. 3.19. Above fill levels of 1 m, the natural sloshing frequencies are approximately constant. This is not the case for the other parameters. Looking at the slosh masses and spring constants, it is clear that only the first slosh mode is of relevance to the system. These modes only have a minor impact for almost empty tanks. However, at this stage, the impact of the sloshing is so small that they are of no importance.

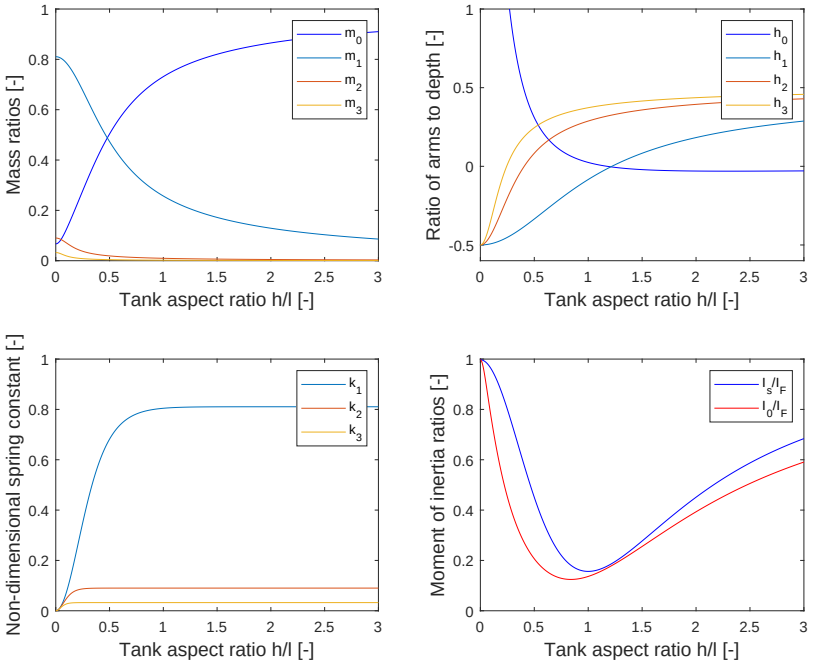


Figure 3.18: Non-dimensional slosh parameters for the first three slosh modes: $n = [1,2,3]$

Table 3.3: Input and output parameters for the propulsion model sensitivity analysis.

Parameter	Value	Unit
Tank length	5.2	m
Tank radius	2	m
Tank volume	65.35	m^3
LO_2 density	1080	$kg\,m^{-3}$
Liquid mass	70,573	kg

There are different ways of modelling the sloshing motion, and its integration with the vehicle dynamics. Nichkawde et al. (2004) conclude that the sloshing dynamics and its coupling with the vehicle motion can be split up into two stages, which removes the inherent coupling between the sloshing and vehicle dynamics. This method is illustrated in Fig. 3.20. Here, the sloshing motion create some sort of disturbance torque in the vehicle dynamics. Then the output of the equations of motion are fed into the sloshing dynamics again. Although this method is fairly straightforward and simple, it does not caption the most important part of the sloshing motion, which is its coupling with the vehicle dynamics. Therefore, a different method is used, where the vehicle and sloshing equations are merged and simulated in one system.

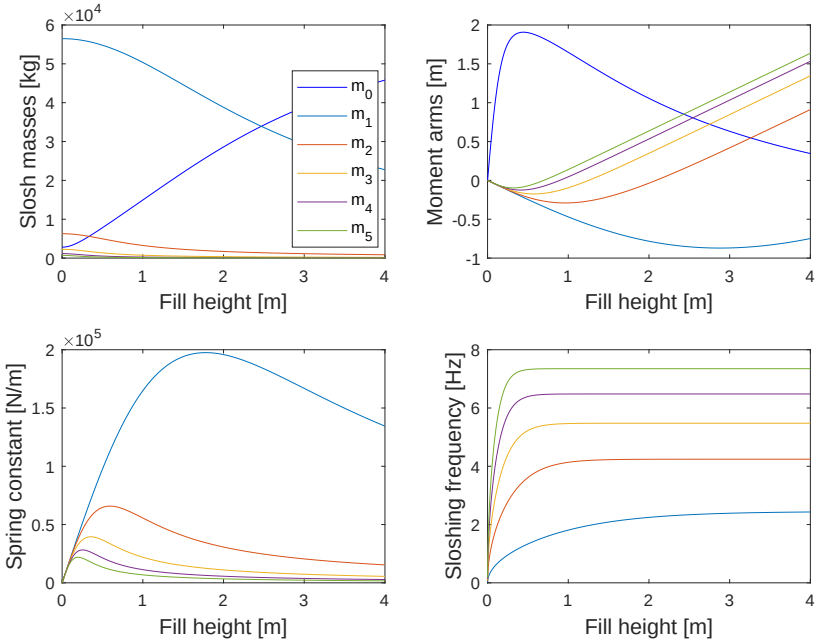


Figure 3.19: Slosh parameters for one of the LO_2 propellant tanks of FSSC-12 with dimensions given in Table 3.3, for the first five slosh modes: $n = [1,2,3,4,5]$

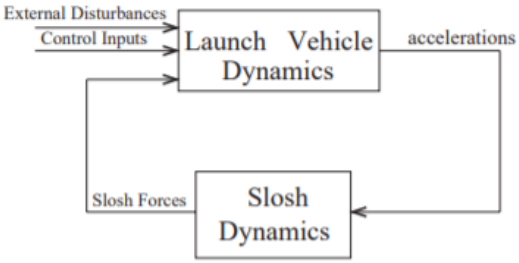


Figure 3.20: Representation of the uncoupled sloshing dynamics system (Nichkawde et al., 2004).

The used equations are derived by Basurto et al. (1966); Raouf (2006) and given as follows:

Force equation

$$m_0\left(\ddot{x} - h_0\ddot{\theta}\right) + \sum_{n=1}^{\infty} m_n\left(\ddot{x}_n + \ddot{x} + h_n\ddot{\theta}\right) = -F_x \tag{3.44}$$

Slosh equation of n^{th} mode

$$m_n\left(\ddot{x}+\ddot{x}_n+h_n\ddot{\theta}\right)+K_n x_n+2m_n\omega_n\zeta_n\dot{x}_n-m_n g\theta=0 \tag{3.45}$$

Moment equation

$$I_0\ddot{\theta}+m_0h_0\left(\ddot{x}-h_0\ddot{\theta}\right)-g\sum_{n=1}^{\infty}m_nx_n+\sum_{n=1}^{\infty}m_nh_n\left(\ddot{x}+\ddot{x}_n+h_n\ddot{\theta}\right)=M_y \tag{3.46}$$

The above equations fully describe the dynamics of the equivalent sloshing model in terms of mass-spring-dashpot systems for translational motion in x-direction and pitching motion. The lateral sloshing is not accounted for in this study. The sloshing state is then given as:

$$\mathbf{x}_{slosh}=(x,x_1,\cdots,x_n,\theta)^T \tag{3.47}$$

with x the position of the vehicle, x_n the slosh mass positions with respect to the vehicle, and θ the pitch angle.

This system cannot be solved in a straightforward way. Hence, they are translated in a different set of equations that resemble a state-space system, with state:

$$\mathbf{x}_{slosh}=\mathbf{x}=\left(x,\dot{x},x_1,\dot{x}_1,\cdots,x_n,\dot{x}_n,\theta,\dot{\theta}\right)^T \tag{3.48}$$

and input matrix:

$$\mathbf{u}=\left(F_x,M_y\right)^T \tag{3.49}$$

The new system of equations in matrix form is given as:

$$\mathbf{E}\dot{\mathbf{x}}=\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{u} \tag{3.50}$$

with:

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & m_0 + \sum_{n=1}^{\infty} m_n & 0 & m_1 & \cdots & 0 & m_n & 0 & -m_0 h_0 + \sum_{n=1}^{\infty} m_n h_n \\ 0 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & m_1 & 0 & m_1 & \cdots & 0 & 0 & 0 & m_1 h_1 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 & 0 \\ 0 & m_n & 0 & 0 & \cdots & 0 & m_n & 0 & m_n h_n \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 \\ 0 & m_0 h_0 + \sum_{n=1}^{\infty} m_n h_n & 0 & m_1 h_1 & \cdots & 0 & m_n h_n & 0 & I_0 - m_0 h_0^2 + \sum_{n=1}^{\infty} m_n h_n^2 \end{bmatrix} \tag{3.51a}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & -K_1 & -2m_1\omega_1\zeta_1 & \cdots & 0 & 0 & m_1 g & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & -K_n & -2m_n\omega_n\zeta_n & m_n g & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 1 \\ 0 & 0 & g m_1 & 0 & \cdots & g m_n & 0 & 0 & 0 \end{bmatrix} \tag{3.51b}$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 \\ -1 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{3.51c}$$

To solve above system of equations, $\mathbf{E}$ can be inverted and moved to the other side:

$$\dot{\mathbf{x}} = \mathbf{E}^{-1}\mathbf{A}\mathbf{x} + \mathbf{E}^{-1}\mathbf{B}\mathbf{u} \tag{3.52}$$

4

FLIGHT ENVIRONMENT

During its ascent, HORUS is subjected to several environments and regimes. Some of the most important factors are outlined in this section, such as the atmospheric conditions, gravity field and wind disturbances. Each of these has an impact on the mechanics of the vehicle during flight and should be assessed to determine how they influence the vehicle.

Section 4.1 describes Earth's gravity field and the gravitational force exerted on the vehicle. In Section 4.2, a simple model to compute the atmospheric properties is outlined. Finally, Section 4.3 details the several forms of wind that act on the vehicle.

4.1. GRAVITY FIELD

The gravitational field of Earth is highly dependent on multiple factors, such as the location and altitude. These parameters, in their turn, depend on the shape of the Earth, which can be called irregular at best. The gravity field depends on the internal mass distribution of the Earth, which can be called irregular at best. However, assuming both the Earth and the vehicle as point masses, the gravitational force between the two masses can be approximated as (Mooij, 2016):

$$\mathbf{F_G} = \frac{GMm}{R^2}\hat{\mathbf{R}} = \frac{\mu m}{R^2}\hat{\mathbf{R}} \tag{4.1}$$

with M and m the masses of the Earth and vehicle respectively, G the universal gravity constant ($G = 6.668 \cdot 10^{-11}$ $\mathrm{m^3\,kg^{-1}\,s^{-1}}$), $\hat{\mathbf{r}}$ the normalised position vector between the point masses, R the distance, and μ the gravitational constant of Earth ($\mu = 3.9860047 \cdot 10^{14}$ $\mathrm{m^3\,s^{-2}}$). By introducing the gravitational potential $U = -\mu/R$, the gravitational force can be rewritten as the gradient of this potential:

$$\mathbf{F_G} = -m\frac{\mu}{R^2}\hat{\mathbf{R}} = -m\nabla U \tag{4.2}$$

The model above is usually called the *central field model*. From Equations (4.1) and (4.2), it is clear that the gravitational force on an object around Earth decreases quadratically with the distance R. For this study of single points in the flight trajectory, the altitude of the vehicle, and therefore gravitational force on the vehicle, remains constant for each simulation. Furthermore, the location of the vehicle above Earth, respectively

its latitude and longitude, are not of interest either. Therefore, the gravitational acceleration can be obtained as follows:

$$g = \frac{g_0 R_0^2}{(R_0 + h)^2} \tag{4.3}$$

where g_0 is the gravitational acceleration at the Earth's surface ($g_0 = 9.80665\ \mathrm{m\,s^{-2}}$) and R_0 is the radial distance at which $g = g_0$ ($R_0 = 6378.137$ km [1]).

4.2. ATMOSPHERE

For common launchers, Earth's atmosphere is but a nuisance that creates drag on the vehicle and increases costs. For spaceplanes, however, the atmosphere can effectively be used to create lift, and use it as an advantage. To have a realistic simulation of the vehicle's motion, the atmospheric properties such as pressure, temperature, density, etc. should be determined. Despite the chaotic nature of the atmosphere, and its complex dynamic processes, there are several available models.

In 1976, a standard atmosphere model was introduced in order to provide consistency in atmospheric calculations (NASA, 1976). The model includes relations to compute density, temperature and pressure. At altitudes lower than 100 km the composition of air is relatively constant or homogeneous. This means the gas constant does not change and the hydrostatic and state equations can easily be integrated. There are two different cases: a layer where the temperature changes according to a defined gradient, and an isothermal layer.

For the first case, the temperature is given by: $T = T_i + \lambda_i(h - h_i)$, with T_i, h_i and λ_i respectively the temperature at the start, the altitude at the start and the lapse rate or temperature gradient of layer i.
The pressure can then be obtained through the integration of the hydrostatic equation:

$$p = p_i \left[\frac{T_i}{T_i + \lambda_i(h - h_i)} \right] \tag{4.4}$$

In an isothermal layer $\lambda = 0$ and the temperature is constant. The pressure is then given by:

$$p = p_i exp\left[-\frac{g}{RT_i}(h - h_i) \right] \tag{4.5}$$

In a perfect gas, the speed of sound a is given as: $a = \sqrt{\gamma RT}$, with γ the adiabatic index or specific heat ratio, which is assumed to be 1.4. Then the Mach number can be computed as follows: $M = V/a$, where V is the flight velocity.

[1] `https://nssdc.gsfc.nasa.gov/planetary/factsheet/earthfact.html` (Last accessed on May 27, 2019)

4.3. WIND DISTURBANCES

Wind is the natural movement of air due to a pressure difference in the atmosphere. Wind is almost always present, in any flight condition, and should therefore be taken into account when studying the stability of a flying vehicle. It generally acts, or at least can be simulated, as a disturbance for the motion of the vehicle.

Viavattene (2018) investigated the impact of different wind forces on the stability and flying qualities of a spaceplane during its ascent. To avoid repetition, this analysis is left out of this study. The analysis by Viavattene (2018) was repeated to verify her conclusions, and it was found that also in this case, the spaceplane can withstand the simulated atmospheric disturbances, namely a steady-state wind, wind gusts and turbulence. The used equations, analysis and results are not given in this report.

There are several reasons why wind does not have a large impact on the motion of the vehicle:

- **Vehicle velocity**: The vehicle is moving at hypersonic velocities, from $M = 5$ to $M = 20$. Its momentum is so large that the wind does not influence the attitude of the vehicle.
- **Vehicle mass**: Similarly to the velocity, the mass of the vehicle increases the momentum of the vehicle, as well as its inertia (resistance to motion). The wind forces are relatively small, and therefore do not have a large impact on the vast vehicle.
- **Altitude**: The atmospheric pressure and density decrease exponentially with altitude. Also the wind velocities and its impact are smaller at higher altitudes. The vehicle travels up to more than 100 km, where the atmosphere is almost non-existent.

5

FLIGHT MECHANICS

The flight behaviour of the vehicle - how it moves and reacts to external and internal forces - is incredibly important. To study the influence of the propulsion system on the vehicle's controllability, it is necessary to first study its attitude dynamics. This study mainly focuses on single points of the vehicle's trajectory, which means the trajectory itself does not need to be simulated. Thus, the translational motion of the vehicle is not taken into account for this study, as the rotational motion alone can already give interesting insights in the motion of the vehicle.

First of all, the necessary reference frames are stated in Section 5.1. Then, Sections 5.2 and 5.3 describe the used state parameters and frame transformations between the previously listed frames. Next, the force models for propulsion and aerodynamics are given in Section 5.4. The kinematic and dynamic equations of motion of the vehicle are derived in Section 5.5. Finally, the inertia model is described in Chapter 5.6.

5.1. REFERENCE FRAMES

Reference frames are generally used to define the state, i.e. position, attitude, etc., of a vehicle with respect to some predefined axes system. Usually, reference frames are represented by three mutually orthogonal axes, which intersect at the origin.

During the ascent phase of the mission, the state of the vehicle needs to be computed. States such as the position and velocity have to be computed and given in a certain reference frame. Different frames are especially useful to simplify the derivation of the equations of motion, as it is often easier to describe a certain motion in one reference frame and then apply a transformation to another. A brief overview of all necessary reference frames is given in Table 5.1. Only the reference frames that are thought to apply to this thesis topic are given. The orientation and notation of the frames are based on the reader by Mooij (2016). Graphical representations of the reference frames are given in Appendix A.

Table 5.1: Reference frame specifications (RF = Reference Frame). The COM column indicates which object is the COM of the RF. The COM of the RF is always the COM of that object.

Name	CoM	Axes	Figure
Body-Fixed RF ($\mathscr{F}_B$)	Vehicle	X_B: In symmetry plane in direction of motion Y_B: Completes right-handed frame Z_B: Downwards in symmetry plane	Fig. B.1, B.2
Trajectory RF ($\mathscr{F}_T$)	Vehicle	X_T: In direction of velocity vector Y_T: Completes right-handed frame Z_T: Downwards in vertical plane	Fig. B.1
Aerodynamic RF ($\mathscr{F}_A$)	Vehicle	X_A: In direction of velocity vector Y_A: Completes right-handed frame Z_A: Opposite to aerodynamic lift force	Fig. B.1
Propulsion RF ($\mathscr{F}_P$)	Vehicle	X_P: In direction of THRUST vector Y_P: Follows from transformation from $\mathscr{F}_B$ frame Z_P: Follows from transformation from $\mathscr{F}_B$ frame	Fig. B.2

5.2. STATE VARIABLES

The motion of a body with respect to a reference frame usually exists of a translational and rotational motion. This motion can be described using mathematical parameters or variables, which are called the state variables. Several sets of variables can be used to describe the motion of a body, each of which can have (dis)advantages to the system. Selecting the state variables is thus quite important, as it can simplify the equations of motion, control laws, and simulation. The full state of the vehicle can be described by its position, velocity, attitude and angular rate.

This study only concerns itself with single points in the trajectory, excluding translational motion. Therefore, these parameters are not included in this report. The core of the model, which was supplied by Mooij and Wijnands (2002), contains the full equations of motions, both rotational and translational. The reports by Mooij and Ellenbroek (2011); Mooij and Wijnands (2002) contain a description of all used equations. Since the equations of motion of the vehicle are already in the simulator, only a summary of the parameters and equations is given here. Derivations are not included.

The rotational motion of the vehicle is expressed in terms of the attitude and angular rates. The attitude is generally thought of as the orientation of the vehicle with respect to some reference frame, whereas the angular rates describe the actual rotation of the vehicle with respect to the reference frame. This section includes all relevant and used parameters related to the attitude or angular rate of the vehicle.

5.2.1. ATTITUDE

The orientation or attitude of an object, or a body-fixed reference frame, with respect to another reference frame can be described in multiple ways, each having certain advantages. This section elaborates on three different definitions, as each of these is used throughout the simulations.

- **Euler angles:** The classical Euler angles - roll angle ϕ, pitch angle θ and yaw angle ψ - are among the most common attitude representations, as they present an intuitive way of representing the attitude of the $\mathscr{F}_B$ frame with respect to the $\mathscr{F}_I$ frame.

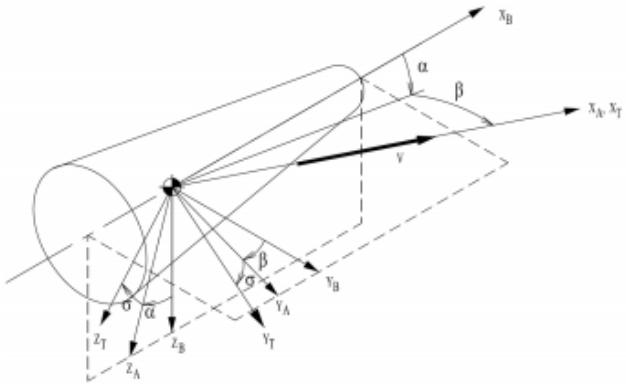


Figure 5.1: Definition of the aerodynamic attitude angles α, β and σ. The three related reference frames are $\mathcal{F}_B$, $\mathcal{F}_A$ and $\mathcal{F}_T$ (Mooij, 2016).

Due to the arbitrary sequence of rotations (maximum three rotations required for ny transformation in three-dimensional space), there are many variations of Euler angles. For aerospace purposes, the 3-2-1 sequence is often used: a yaw rotation about the Z-axis, a pitch rotation about the Y-axis, and finally a roll rotation about the X-axis (Mooij, 2016; Mulder et al., 2013). A special form of Euler angles are given in the next point.

- **Aerodynamic angles:** For control studies, the aerodynamic angles provide a convenient way of representing the attitude. Note that these are technically also Euler angles and also a set of three angles: the angle of attack α ($-180° \leq \alpha < 180°$), angle of sideslip β ($-90° \leq \beta \leq 90°$), and bank angle σ ($-180° \leq \sigma < 180°$). These angles in their positive direction are shown in Fig. 5.1.

- **Quaternions:** Whereas the former sets provide an intuitive way of describing an attitude, quaternions are quite the opposite. A quaternion $\mathbf{q} = (q_1, q_2, q_3, q_4)$ is a 4-dimensional hyper-complex number, consisting of one real and three imaginary numbers. Quaternions have the advantage of not experiencing any singularities, as opposed to Euler angles.

As mentioned before, the angle representations provide an intuitive way of representing the attitude of a body. It is relatively straightforward to imagine a rotation as the sequence of three rotations about certain axes with specified angles. This certainly facilitates the physical interpretation and visualisation of the rotational motion and attitude of the body. However, these sets do experience singularities, since they do not contain any redundant parameters. The roll, pitch and yaw Euler angles experience a singularity at $\theta = \pm 90°$, and the aerodynamic angles at $\beta = \pm 90°$. For this study, both of these situations should not occur during the ascent flight. Therefore, quaternions are not required, and the intuitive angles were chosen for most of the control laws and other equations that are developed in this thesis. Note that quaternions are used in the GAOCS simulator that is used as the core to build upon.

Table 5.2: Frame transformations and their corresponding parameters. The full matrices are given in Appendix B.

Frames	Transformation sequence	Parameters		
$\mathscr{F}_B \rightarrow \mathscr{F}_A$	$\mathbf{C}_{A,B} = \mathbf{C}_Z(\beta)\mathbf{C}_Y(-\alpha)$	β:	Sideslip angle	[rad]
		α:	Angle of attack	[rad]
$\mathscr{F}_A \rightarrow \mathscr{F}_T$	$\mathbf{C}_{T,A} = \mathbf{C}_X(\sigma)$	σ:	Bank angle	[rad]
$\mathscr{F}_B \rightarrow \mathscr{F}_P$	$\mathbf{C}_{P,B} = \mathbf{C}_Y(\epsilon_T)\mathbf{C}_Z(\psi_T)$	ϵ_T:	Thrust elevation angle	[rad]
		ψ_T:	Thrust azimuth angle	[rad]

For attitude control, the aerodynamic angles are used. The reason is that controllability and stability are mainly related to aerodynamic forces and moments, which are easily defined in the aerodynamic reference frames.

5.2.2. ANGULAR RATES

The angular rate is the rotational velocity of the body-fixed frame with respect to the inertial frame, expressed in components along the body axes: X_B, Y_B and Z_B. The angular rate $\boldsymbol{\omega}$ is therefore defined by the roll rate p, the pitch rate q, and yaw rate r:

$$\boldsymbol{\omega} = \left(p, q, r\right)^T \tag{5.1}$$

5.3. FRAME TRANSFORMATIONS

It is often necessary to transform elements described in one coordinate system into another one. This could be because an aerodynamic force given in the aerodynamic frame must be represented in the body frame. In this section, the most relevant coordinate transformations are given. It is assumed the reader has the necessary background to understand the basic theory of frame transformations to keep things concise.

Table 5.2 contains all base transformations which are used throughout this study. It should be noted that other transformations can be done using a subsequent set of the base transformations. For example, the transformation from $\mathscr{F}_B$ to $\mathscr{F}_{TA}$ can be obtained through the combination of: $\mathbf{C}_{TA,B} = \mathbf{C}_{TA,AA}\mathbf{C}_{AA,B}$. The full transformation matrices are given in Appendix B.

5.4. FORCE MODELS

In Section 5.5.1, the dynamic rotational equations of motion were stated. These included a moment vector which was not yet defined. This moment vector includes two main contributions: aerodynamic and propulsion moments. This section elaborates on each of these and states how they can be computed in the body frame.

5.4.1. AERODYNAMIC MOMENTS

The aerodynamic moments $\mathbf{M}_{a_m,B}$ in the body frame are defined as

$$\mathbf{M}_{a_m,B} = \begin{pmatrix} \mathscr{L} \\ \mathscr{M} \\ \mathscr{N} \end{pmatrix} = \begin{pmatrix} \mathbf{C}_l \bar{q} S_{ref} b_{ref} \\ \mathbf{C}_m \bar{q} S_{ref} c_{ref} \\ \mathbf{C}_n \bar{q} S_{ref} b_{ref} \end{pmatrix} \tag{5.2}$$

Here, $\mathscr{L}, \mathscr{M}$ and $\mathscr{N}$ are the rolling, pitching and yawing moments, respectively. b_{ref}, c_{ref} and S_{ref} are the aerodynamic reference lengths and surface, the wing span, chord length and surface area, which are geometric properties of the vehicle. Together with the dynamic pressure $\bar{q}$ given in Eq. (5.3), where ρ is the local atmospheric density and A the airspeed, these variables can be multiplied with the dimensionless moment components $\mathbf{C}_i$. The dimensionless coefficients of the vehicle are usually determined through rigorous modelling and/or wind tunnel tests.

$$\bar{q} = \frac{1}{2}\rho A^2 \tag{5.3}$$

Now, the contribution of the aerodynamic forces $\mathbf{F}_{a,AA}$, given below in the aerodynamic frame, on the total moment should be determined.

$$\mathbf{F}_{a,AA} = \begin{pmatrix} -D \\ -S \\ -L \end{pmatrix} = \begin{pmatrix} -\mathbf{C}_D \bar{q} S_{ref} \\ -\mathbf{C}_S \bar{q} S_{ref} \\ -\mathbf{C}_L \bar{q} S_{ref} \end{pmatrix} \tag{5.4}$$

The aerodynamic forces are the lift L, drag D and side force S. As they are defined in the aerodynamic reference frames, there is a minus sign in front of each component. It is interesting to define these forces in the body frame. The transformation matrix , $\mathbf{C}_{AA,B}$, from Eq. (B.1), describes the relation between both frames. Instead of lift, drag and side force, the components are then given by X, Y and Z, as shown in Eq. (5.5). These forces act in the aerodynamic centre, which is not necessarily coinciding with the COM.

$$\mathbf{F}_{a,B} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} \mathbf{C}_X \bar{q} S_{ref} \\ \mathbf{C}_Y \bar{q} S_{ref} \\ \mathbf{C}_Z \bar{q} S_{ref} \end{pmatrix} \tag{5.5}$$

The location of the COM with respect to the aerodynamic centre is given as $\mathbf{r}_{cm} = \{x_{cm}, y_{cm}, z_{cm}\}^T$. Now the moment due to the aerodynamic forces results from the vector multiplication:

$$\mathbf{M}_{a_f,B} = \mathbf{r}_{cm} \times \mathbf{F}_{a,B} \tag{5.6}$$

Then the total moment vector is given as follows:

$$\mathbf{M}_{a,B} = \mathbf{M}_{a_f,B} + \mathbf{M}_{a_m,B} \tag{5.7}$$

5.4.2. PROPULSION MOMENTS

Propulsion forces usually do not act through the COM of the vehicle, creating a moment around the COM. In certain applications, this is even used as a control measure, called thrust vector control. With this method, the thrust vector is a control parameter which can be used to change the attitude of the vehicle to a more favourable orientation.

To compute the propulsion moments, the thrust force $\mathbf{F}_T$ and its direction need to be known, preferably in the body frame. The thrust force in the propulsion frame is simply $\mathbf{F}_{T,P} = (T, 0, 0)^T$, with T computed using the methods from Chapter 3. Using the inverse transformation $\mathbf{C}_{P,B}^{-1}$ from Section 5.3, the thrust force in the body frame $\mathbf{F}_{T,B} =$

$(T_x, T_y, T_z)^T$, in Cartesian components along the body axes, can be determined as given in Eq. (5.8). Here, T is the magnitude of the thrust force.

$$T_x = T\cos\psi_T\cos\epsilon_T \tag{5.8a}$$
$$T_y = T\sin\psi_T\cos\epsilon_T \tag{5.8b}$$
$$T_z = -T\sin\epsilon_T \tag{5.8c}$$

The thrust force direction is determined by the angles ψ_T and ϵ_T, which can be adapted to change the attitude of the vehicle during flight. This thrust vector control is mainly used during the ascent phase where the aerodynamic control surfaces are not efficient due to low atmospheric density.

The thrust moment can then be obtained from the cross product:

$$\mathbf{M}_{T,B} = \mathbf{r}_T \times \mathbf{F}_{T,B} = F_T \begin{pmatrix} -y\sin\epsilon_T - z\sin\psi_T\cos\epsilon_T \\ z\cos\psi_T\cos\epsilon_T + x\sin\epsilon_T \\ x\sin\psi_T\cos\epsilon_T - y\cos\psi_T\cos\epsilon_T \end{pmatrix} \tag{5.9}$$

with $\mathbf{r}_T = (x, y, z)$ the location vector of the engines. Usually, the engines cannot move extensively, which means the thrust angles always remain quite small. Hence, the small angle approximation ($\cos\alpha = 1$ and $\sin\alpha = \alpha$) can be used to simplify above equation:

$$\mathbf{M}_{T,B} = \mathbf{r}_T \times \mathbf{F}_{T,B} = F_T \begin{pmatrix} -y\epsilon_T - z\psi_T \\ z + x\epsilon_T \\ x\psi_T - y \end{pmatrix} \tag{5.10}$$

This relation can now be used for TVC. This method relies on the inversion of above equation to find a combination of ϵ_T and ψ_T that can deliver a required control moment. From Eq. (5.10), it is clear that this system is over-determined, since there are only two variables for three equations. Hence, it is not always possible to find a solution that satisfies all equations. The pitch and yaw moments can independently be satisfied, but the roll moment introduces the issue. This problem can be solved in several ways:

- Create a dedicated control allocation algorithm that will determine which of the combinations of ϵ_T and ψ_T is optimal for a certain commanded moment vector. This method tries to find a solution for the control problem by only using one engine.
- Adding more than one engine might solve this problem, since adding an engine adds two more variables to solve the same equations. Of course, this adds complexity to both the simulation models, as well as the physical system, as both engines should be able to move independently in this case. The system would still need some sort of control allocation algorithm in this case to determine the thrust angles for all engines, which gives an optimal solution to the problem.
- Use the main engines for pitch and yaw control, and control the roll moments with an independent set of roll thrusters. Thus is a simple and robust method to solve the problem, since the roll motion is the issue in the problem. By taking it out of

the TVC problem, the problem is effectively mitigated. Roll thrusters are already present on the vehicle, since they are used during re-entry.

For this study, the third option is chosen. It is robust and does not introduce unnecessary complications in the models or physical system. Note that this method has a drawback as well. The main engines will introduce an extra roll moment by changing their thrust angles. This thrust moment can be in the correct direction, and help the roll thrusters control the vehicle, but the opposite case is possible as well, where the engines create a roll moment in the undesired direction. The thrusters will then have to compensate for this moment as well. The main engines are usually located quite close to the COM of the vehicle in y and z direction, so the induced roll moment will be small in any case. The roll moment of the engines is added to the system as a disturbance torque. The thrust angles are obtained as follows:

$$\epsilon_T = \frac{M_y - zF_T}{xF_T} \tag{5.11a}$$

$$\psi_T = \frac{M_z + yF_T}{xF_T} \tag{5.11b}$$

If one of the angles exceeds its limit, the angle is set equal to the limiting angle.

Each engine has its separate influence on the moment, so to be accurate, the total moment should be taken as the sum of the contribution of each engine. Therefore, the moment of engine i, located at $\mathbf{r}_{T,i}$ is given as

$$\mathbf{M}_{T,B_i} = \mathbf{r}_{T,i} \times \mathbf{F}_{T,B_i} \tag{5.12}$$

and the total moment, including n engines, is then

$$\mathbf{M}_{T,B} = \sum_{i=1}^{n} \mathbf{M}_{T,B_i} \tag{5.13}$$

5.5. EQUATIONS OF MOTION

To model the vehicle, its equations of motion (EOM) need to be determined. The motion of the vehicle can be separated into a translation of its centre of mass and a rotation about its centre of mass. Here, only the rotational aspect is taken into account as the translational behaviour of the vehicle is outside the scope of this study. Since the main equations of motion of the vehicle are not developed specifically for this thesis, but are already present in the used GAOCS simulator, only a summary of the equations is given here, without any derivations.

5.5.1. DYNAMICS

For this analysis, only a set of points will be analysed, which means the mass of the body at those points is assumed constant. Thus, both the relative and Coriolis moments are zero in this case. Moreover, the vehicle is also assumed to be non-elastic, leading to a simplified equation of motion given in Eq. (5.14), also called the Euler equations.

$$\dot{\boldsymbol{\omega}} = \mathbf{I}^{-1}\left(\tilde{\mathbf{M}}_{cm} - \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega}\right) \tag{5.14}$$

$\tilde{\mathbf{M}}_{cm}$ contains all external moments acting on the vehicle. These originate from aerodynamic forces such as lift and drag, the thrust force and control moments from the aerodynamic control surfaces such as elevators, ailerons and rudders. All these forces can create moments around the vehicle's centre of mass and will thus have an influence on the motion of the spaceplane. In the body-fixed reference frame, the moment vector can be written as: $\tilde{\mathbf{M}}_{cm} = \{M_x, M_y, M_z\}^T$, where each component M_i represents the moment along one of the body axes.

I is the inertia tensor as given in Eq. (5.15).

$$\mathbf{I} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \tag{5.15}$$

5

In certain cases, a symmetry assumptions can be made to simplify the inertia matrix. As the vehicle is approximately symmetric along the $X_B Z_B$-plane, all inertia cross-products involving y can be assumed to be negligibly small in comparison with the other inertia values, leading to the following inertia matrix:

$$\mathbf{I} = \begin{bmatrix} I_{xx} & 0 & -I_{xz} \\ 0 & I_{yy} & 0 \\ -I_{xz} & 0 & I_{zz} \end{bmatrix} \tag{5.16}$$

However, it should be noted that this is an assumption that can only be made in certain cases. The original inertia tensor will still be used in the main part of the report, as it is of interest to study the influence of mass asymmetries on the stability and control of the vehicle. During flight, the inertia and mass of the vehicle change due to mass expulsion and other small factors. This of course does have an influence on the stability of the spaceplane.

The full dynamic equations of motion in matrix form are then given in Eq. (5.17).

$$\begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \begin{pmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{pmatrix} = \begin{pmatrix} M_x \\ M_y \\ M_z \end{pmatrix} - \begin{pmatrix} p \\ q \\ r \end{pmatrix} \times \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \tag{5.17}$$

The inverse of the inertia matrix is given by the following expression (Mooij, 1998):

$$\mathbf{I}^{-1} = \frac{1}{I^{**}} \begin{bmatrix} I_{yy}I_{zz} - I_{yz}^2 & I_{xz}I_{yz} + I_{xy}I_{zz} & I_{xy}I_{yz} + I_{xz}I_{yy} \\ I_{xz}I_{yz} + I_{xy}I_{zz} & I_{xx}I_{zz} - I_{xz}^2 & I_{xy}I_{xz} + I_{yz}I_{xx} \\ I_{xy}I_{yz} + I_{xz}I_{yy} & I_{xy}I_{xz} + I_{yz}I_{xx} & I_{xx}I_{yy} - I_{xy}^2 \end{bmatrix} \tag{5.18}$$

with

$$I^{**} = \frac{1}{I_{xx}I_{yy}I_{zz} - 2I_{xy}I_{xz}I_{yz} - I_{yz}^2 I_{xx} - I_{xz}^2 I_{yy} - I_{xy}^2 I_{zz})} \tag{5.19}$$

The moment vector $\{M_x, M_y, M_z\}^T$ can be further evaluated to include the separate contributions of the different acting forces. The gravitational force is taken to be uniformly distributed, which means it will act in the vehicle's centre of mass, and not create a moment around it. Section 5.4 gives expressions for the aerodynamic and propulsion moments in the body frame: $\mathbf{M}_{a,B}$ and $\mathbf{M}_{T,B}$ ($\mathbf{M}_a$ and $\mathbf{M}_T$ in short). The updated moment vector is given by:

$$\tilde{\mathbf{M}}_{cm} = \begin{pmatrix} M_{a,x} + M_{T,x} \\ M_{a,y} + M_{T,y} \\ M_{a,z} + M_{T,z} \end{pmatrix} \tag{5.20}$$

5.5.2. KINEMATICS

For the aerodynamic angles, the kinematics of the system consist of three equations, one for each of the attitude angles α, β and σ. The full kinematic attitude equations are given in the report by (Mooij, 1998), but are not repeated here, since these full equations can be simplified when only rotational motion at single points is analysed. The derivations of these equations are not included in the report as they are not relevant at this point. Assuming a single point, the kinematic equations can be given as follows:

$$\begin{pmatrix} p \\ q \\ r \end{pmatrix} = \begin{bmatrix} 0 & \sin\alpha_g & -\cos\beta\cos\alpha \\ 1 & 0 & -\sin\beta \\ 0 & -\cos\alpha & -\cos\beta\sin\alpha \end{bmatrix} \begin{pmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\sigma} \end{pmatrix} \tag{5.21}$$

$$\begin{pmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\sigma} \end{pmatrix} = \frac{1}{\cos\beta} \begin{bmatrix} -\cos\alpha\sin\beta & \cos\beta & -\sin\alpha\sin\beta \\ \sin\alpha\cos\beta & 0 & -\cos\alpha\cos\beta \\ -\cos\alpha & 0 & -\sin\alpha \end{bmatrix} \begin{pmatrix} p \\ q \\ r \end{pmatrix} \tag{5.22}$$

where 'sin' and 'cos' represent *sine* and *cosine* functions.

Clearly, there is a singularity in the above equations when $\cos\beta = 0$ or $\beta = 90$ deg. A sideslip angle of 90 degrees should not occur during the simulations, except perhaps at standstill, so this does not result in an issue.

5.6. VEHICLE INERTIA

The inertia of the spaceplane varies continuously throughout its ascent flight to space. This is due to the expulsion of propellants, and the motion of the propellants in the tanks. Motion of propellant tanks has two effects on inertia. Firstly, the propellant moves to stay level with respect to the gravity vector. To study this, the effect of angle of attack on propellant inertia is added in the model. Changes in bank or sideslip angle are not considered, as these angles do not vary as much during the ascent phase, and therefore have a minor influence on the inertia. A secondary effect is the changing inertia due to sloshing. Due to time constraints, sloshing was not investigated in depth, but rather on a high-level basis. Its impact on changing inertia is therefore left out of this study.

The inertia change due to an inclination angle for separate tanks was analysed in Section 3.3. For this study, the vehicle is assumed to consist of a fixed dry or empty mass with constant inertia, and the moving propellant masses. The model of a single tank

can therefore easily be added to the system. If a vehicle has, for example, five tanks, the model for one tank can be copied five times, of course with other dimensions and locations conforming to the size of the individual tanks. Together, the separate tanks, and the fixed empty mass make up the complete vehicle.

To compute the COM and MMOI of the complete vehicle at any point in the vehicle's trajectory, Equations (5.23) and (5.24) can be used, respectively. The summation in the equations is always over the number of tanks, which can be varied in the model. m_{tank} and m_{dry} are the mass of the propellant in a single tank and the total mass of the dry vehicle, respectively. The x, y and z components below indicate the distance of the centre of mass of a certain component (dry mass or propellant) in a predetermined body reference frame.

$$x_{com} = \frac{\sum(m_{tank}x_{tank}) + m_{dry}x_{dry}}{\sum m_{tank} + m_{dry}} \tag{5.23a}$$

$$y_{com} = \frac{\sum(m_{tank}y_{tank}) + m_{dry}y_{dry}}{\sum m_{tank} + m_{dry}} \tag{5.23b}$$

$$z_{com} = \frac{\sum(m_{tank}z_{tank}) + m_{dry}z_{dry}}{\sum m_{tank} + m_{dry}} \tag{5.23c}$$

$$I_{xx} = \sum\left(I_{xx_{tank}} + m_{tank}\left(d^2_{y_{tank}} + d^2_{z_{tank}}\right)\right) + \left(I_{xx_{dry}} + m_{dry}\left(d^2_{y_{dry}} + d^2_{z_{dry}}\right)\right) \tag{5.24a}$$

$$I_{yy} = \sum\left(I_{yy_{tank}} + m_{tank}\left(d^2_{x_{tank}} + d^2_{z_{tank}}\right)\right) + \left(I_{yy_{dry}} + m_{dry}\left(d^2_{x_{dry}} + d^2_{z_{dry}}\right)\right) \tag{5.24b}$$

$$I_{zz} = \sum\left(I_{zz_{tank}} + m_{tank}\left(d^2_{x_{tank}} + d^2_{y_{tank}}\right)\right) + \left(I_{zz_{dry}} + m_{dry}\left(d^2_{x_{dry}} + d^2_{y_{dry}}\right)\right) \tag{5.24c}$$

$$I_{xy} = \sum\left(I_{xy_{tank}} + m_{tank}d_{x_{tank}}d_{y_{tank}}\right) + \left(I_{xy_{dry}} + m_{dry}d_{x_{dry}}d_{y_{dry}}\right) \tag{5.24d}$$

$$I_{xz} = \sum\left(I_{xz_{tank}} + m_{tank}d_{x_{tank}}d_{z_{tank}}\right) + \left(I_{xz_{dry}} + m_{dry}d_{x_{dry}}d_{z_{dry}}\right) \tag{5.24e}$$

$$I_{yz} = \sum\left(I_{yz_{tank}} + m_{tank}d_{y_{tank}}d_{z_{tank}}\right) + \left(I_{yz_{dry}} + m_{dry}d_{y_{dry}}d_{z_{dry}}\right) \tag{5.24f}$$

Now that the inertia and COM of the vehicle, including its moving propellant, can be computed, there is one more aspect that should be added to the model, i.e. in what sequence the tanks are being emptied. Daimler-Benz Aerospace (1996c) argue that there are two main options: all tanks are emptied at the same time, or there is a certain predetermined sequence, where one tank remains full until another is empty. These two options are added to the model, neglecting intermediate cases, where, for example, one tank would be emptied till it is half full, after which another is emptied and so on. To treat all of these cases is both out of the scope of this study, as well as not very practical in reality. The main advantage of a dedicated 'emptying' sequence could be to limit the change in COM and inertia throughout the flight. Another assumption is made that all tanks of the same type (either oxidator or fuel) have equal mass flow Daimler-Benz Aerospace (1996c).

The model works as outlined below:

1. **Set inputs:** Tank dimensions, OF ratio, mass already expelled and emptying sequence. The used mass can be computed at any point in time by multiplying the mass flow with the time. If the mass flow is not constant, the area under the mass flow curve can be computed to find the expelled mass. The emptying sequence is a list of integers indicating which tanks are emptied in what order. As an example, consider five tanks (T_1, T_2, T_3, T_4, T_5), where T_1, T_2 and T_3 are oxygen tanks, and T_4 and T_5 fuel tanks (e.g. with LH_2). If all tanks are emptied at the same time, the input sequence is given as *[1,2,3],[4,5]*. Tanks being emptied at the same time are put between brackets, and oxygen and fuel tanks are always kept separate. For the case where T_1 and T_2 are emptied before T_3, and T_5 before T_4, the sequence is given as *[1,2],3,5,4*.

2. **Determine used fuel and oxidator:** Using the OF and total used mass, the amount of oxygen and fuel can be obtained as: $m_{fuel} = \frac{1}{OF+1} m_{used}$, $m_{ox} = \frac{OF}{OF+1} m_{used}$.

3. **Compute leftover mass in each tank:** In order of the input sequence, the used oxidator or fuel mass is subtracted from the available mass in the corresponding tank. If there is still mass left (e.g. there is more mass in the tank, than mass already used), nothing happens. Otherwise, the next tank in the sequence is checked, until all the used mass is divided over the tanks, or all tanks are empty.

Fig. 5.2 demonstrates the example given above. All tanks are emptied at the same time at the start as expected with the sequence *[1,2,3],[4,5]*. At a certain point, the smaller tanks (OX1, OX2 and H2) are empty, after which all the expelled mass is taken from the remaining tanks. This is visible through the increased inclination of the lines for OX3 and H2 after the other tanks are empty. Section 7.3.3 shows some more results in order to verify this algorithm.

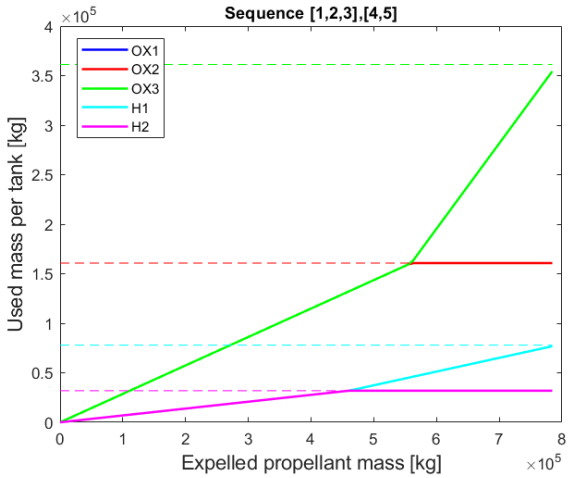


Figure 5.2: Used mass per tank w.r.t. total expelled mass for a vehicle with five tanks with tank sequence "[1,2,3],[4,5]". The tank dimensions are taken from the FSSC-1 vehicle (Daimler-Benz Aerospace, 1996c).

6

CONTROL SYSTEM

Control systems are a key part of any flight vehicle. Without these, the stability of the craft cannot be guaranteed, and performing manoeuvres becomes an impossible task for the pilot or computer. The general task of the controller is to compute the necessary 'actions' that need to be performed to attain or reach a desired orientation, attitude, or more generally, state. In this case, the controller supplies the model with the required control moments or deflections of the control surfaces.

The LQR controller, described in Appendix D, which is used as a benchmark for the other controllers developed in this study, is a linear controller, based on a linearised system. These inherently cannot handle nonlinear systems very well, and are usually not as accurate as their nonlinear counterparts. It was therefore decided to implement a nonlinear controller, which better suits the needs of this project. Because of the nonlinearity of the equations of motion, nonlinear dynamic inversion (NDI) control was chosen. It should perform better and not diverge as early as the LQR controller.

The interested reader is referred to for a detailed description of the theory on NDI, its derivations and applications. Three different versions of NDI: normal, with time scale separation, and an incremental form, are outlined in Sections 6.1 to 6.3. The actuator assignment algorithm is summarised in Section 6.4.

6.1. BASIC NDI CONTROLLER FOR HORUS

NDI is used with nonlinear systems and uses a virtual controller. This method has two control loops. The inner loop linearises the system and the outer one controls the linearised system, making use of a simple PD controller. The method is mainly based on the work of Chu (2018). In order to have a more convenient representation of the NDI equations, a different form of the equations of motion, given in Equations (5.17) and (5.22), is required.

The state derivative is obtained as:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} \tag{6.1}$$

with

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \mathbf{N}\boldsymbol{\omega} \\ -\mathbf{I}^{-1}\boldsymbol{\Omega}\mathbf{I}\boldsymbol{\omega} \end{bmatrix} \quad , \quad \mathbf{G}(\mathbf{x}) = \begin{bmatrix} 0_{3\times3} \\ \mathbf{I}^{-1} \end{bmatrix} \tag{6.2}$$

The inner control loop is given in the equation below:

$$\mathbf{u} = \mathbf{M}(\mathbf{x})^{-1}(\mathbf{v} - \mathbf{l}(\mathbf{x})) \tag{6.3}$$

where $\mathbf{v}$ is the virtual controller given by Eq. (6.4). $\mathbf{M}(x)$ and $\mathbf{l}(x)$ are given in Eq. (6.5). These matrices are obtained using the Lie derivative of the output variables until the control variable $\mathbf{u}$ is present in the equations (Chu, 2018). This is the core of the NDI method, as at this point there is a direct relation between the in- and outputs. $\mathbf{N}$ is the attitude matrix, given by Eq. (6.6), and Ω is the attitude rate skew matrix. The control vector $\mathbf{u}$ is given as: $\mathbf{u} = (\eta_x \mathscr{M}_{max}, \eta_y \mathscr{L}_{max}, \eta_z \mathscr{N}_{max})^T$. The reference attitude rates $\boldsymbol{\omega}_{ref}$ are zero. $\boldsymbol{\theta}$ are the aerodynamic angles: $\boldsymbol{\theta} = (\alpha, \beta, \sigma)^T$

$$\mathbf{v} = -\mathbf{K}_P\left(\boldsymbol{\theta} - \boldsymbol{\theta}_{ref}\right) - \mathbf{K}_D\left(\boldsymbol{\omega} - \boldsymbol{\omega}_{ref}\right) \tag{6.4}$$

$$\mathbf{M}(\mathbf{x}) = \tfrac{\partial}{\partial x}\left[\mathbf{N}\boldsymbol{\omega}\right]\begin{bmatrix} 0_{3\times3} \\ \mathbf{I}^{-1} \end{bmatrix} \quad , \quad \mathbf{l}(\mathbf{x}) = \tfrac{\partial}{\partial x}\left[\mathbf{N}\boldsymbol{\omega}\right]\begin{bmatrix} \mathbf{N}\boldsymbol{\omega} \\ -\mathbf{I}^{-1}\boldsymbol{\Omega}\mathbf{I}\boldsymbol{\omega} \end{bmatrix} \tag{6.5}$$

$$\mathbf{N} = \frac{1}{\cos\beta}\begin{bmatrix} -\cos\alpha\sin\beta & \cos\beta_g & -\sin\alpha\sin\beta \\ \sin\alpha\cos\beta & 0 & -\cos\alpha\cos\beta \\ -\cos\alpha & 0 & -\sin\alpha \end{bmatrix} \tag{6.6}$$

The partial derivative of the kinematic equation is given as follows:

$$\frac{\partial}{\partial x}\left[\mathbf{N}\boldsymbol{\omega}\right] = \begin{bmatrix} \mathrm{s}\alpha\mathrm{t}\beta p - \mathrm{c}\alpha\mathrm{t}\beta r & -\frac{\mathrm{c}\alpha p + \mathrm{s}\alpha r}{\mathrm{c}\beta^2} & 0 & -\mathrm{c}\alpha\mathrm{t}\beta & 1 & -\mathrm{s}\alpha\mathrm{t}\beta \\ \mathrm{c}\alpha p + \mathrm{s}\alpha r & 0 & 0 & \mathrm{s}\alpha & 0 & -\mathrm{c}\alpha \\ \frac{\mathrm{s}\alpha p - \mathrm{c}\alpha r}{\mathrm{c}\beta} & -\frac{\mathrm{c}\alpha\mathrm{t}\beta p + \mathrm{s}\alpha\mathrm{t}\beta r}{\mathrm{c}\beta} & 0 & -\frac{\mathrm{c}\alpha}{\mathrm{c}\beta} & 0 & -\frac{\mathrm{s}\alpha}{\mathrm{c}\beta} \end{bmatrix} \tag{6.7}$$

The gains are determined through trial and error, by changing their value across multiple simulations to find an appropriate value.

6.2. NDI CONTROLLER WITH TIME SCALE SEPARATION

The next method is the Time Scale Separation NDI or TSS method. A property of flight systems is that some of the dynamics are faster than the other. Variables have slow dynamics when the control effectiveness on their dynamics is low. So when the control effectiveness is high, the dynamics are said to be fast (Chu, 2018). In other words: when a change is made in one of the controls, variables with fast dynamics will react almost immediately, whereas the ones with slow dynamics will not show an immediate reaction. For flight control, this separation is usually between the kinematics and dynamics of the system, e.g. the angles and rates. NDI is applied twice to each of the separate systems, one of which is the inner loop, and the other the outer. The input computed by the outer loop NDI is the reference signal to the inner loop NDI. The inner loop consists of the fast dynamics, and vice versa.

The main process of NDI is the same as described in the previous sections, and is not repeated here. The TSS algorithms are directly applied to HORUS in Section 6.2. The first inversion is given as:

$$\mathbf{u}_{out} = \boldsymbol{\omega}_{ref} = \mathbf{N}^{-1}\mathbf{v}_{\theta} = \mathbf{N}^{-1}\mathbf{K}_p\left(\boldsymbol{\theta}_{ref} - \boldsymbol{\theta}\right) \tag{6.8}$$

The inner loop consists of the dynamic equations of motion. Here, the output of the outer loop, ω_{ref}, is used. The required control torque is then given as:

$$\mathbf{u}_{in} = (\eta_x \mathscr{M}_{max}, \eta_y \mathscr{L}_{max}, \eta_z \mathscr{N}_{max})^T = \mathbf{I}\mathbf{v}_w + \boldsymbol{\Omega}\mathbf{I}\boldsymbol{\omega} \tag{6.9}$$

with $\mathbf{v}_{\omega} = \mathbf{K}_d(\omega_{ref} - \omega)$. Clearly, in this case: $\mathbf{M}(\mathbf{x}) = \mathbf{I}^{-1}$ and $\mathbf{l}(\mathbf{x}) = -\mathbf{I}^{-1}\boldsymbol{\Omega}\mathbf{I}\boldsymbol{\omega}$.

6.3. INCREMENTAL NDI CONTROLLER

Incremental NDI or in short INDI, is another form of NDI, which has certain advantages in comparison with the previous methods. It is similar to the TSS method. The key difference between the two methods is that INDI implements a sensor which measures the angular accelerations instead of taking the rates from a predetermined model of the system. Because of this approach, an inaccurate dynamic model does not influence the controller as much. This is certainly interesting for systems that have several uncertainties, or that are very complicated. Since the motion of HORUS depends on many factors, such as the atmospheric conditions, thrust, aerodynamics, etc. the model of the system will never be extremely accurate. By measuring the angular rate during the simulation, these inaccuracies are effectively ignored and even taken out of the loop. Of course, the used sensor will have errors, noise and biases, which leads to inaccurate measurements. This dependency on the sensor is the main drawback of this approach. The INDI method is described below.

First, the dynamics are approximated using a 1st order Taylor series expansion:

$$\dot{\boldsymbol{\omega}} = \dot{\boldsymbol{\omega}}_0 + \frac{\delta\dot{\boldsymbol{\omega}}}{\delta\mathbf{u}}\Delta\mathbf{u} + \frac{\delta\dot{\boldsymbol{\omega}}}{\delta\boldsymbol{\omega}}\Delta\boldsymbol{\omega} \tag{6.10}$$

Ignoring the third term on the right, and using $\frac{\delta\dot{\boldsymbol{\omega}}}{\delta\mathbf{u}} = \mathbf{I}^{-1}$ simplifies above equation (Chu, 2018) to form:

$$\dot{\boldsymbol{\omega}} = \dot{\boldsymbol{\omega}}_0 + \mathbf{I}^{-1}\Delta\mathbf{u} \tag{6.11}$$

This equation is a lot more simple than the dynamic equations from previous methods. This is because the sensors provide much of the necessary information, and thus make certain components in the equations obsolete. All of this information is contained in the term $\dot{\boldsymbol{\omega}}_0$.

Now, NDI needs to be applied to the EOM for HORUS. First of all, the control moment increment is determined as $\Delta\mathbf{u} = \mathbf{I}(\mathbf{v}_w - \dot{\boldsymbol{\omega}}_0)$. Here, $\mathbf{v}_w$ is the virtual controller obtained in the same way as explained in the previous chapter about the TSS method. The necessary relations are given in Eq. (6.12).

$$\begin{aligned} \omega_{\mathbf{ref}} &= \mathbf{K}(\theta)^{-1}(\mathbf{v}_\theta - \mathbf{K}(\theta)_g) \\ \mathbf{v}_\theta &= \mathbf{K}_p(\theta_{ref} - \theta) \\ \mathbf{v}_\omega &= \mathbf{K}_d(\omega_{ref} - \omega) \end{aligned} \tag{6.12}$$

For the nominal simulations, the measurement of $\dot{\boldsymbol{\omega}}_0$ is assumed to be perfect. This means the sensor is measuring the exact change in angular rate, without the addition of any noise or biases. An imperfect sensor measurement will be investigated as well, but more separate from the other simulations, mainly as an addition for completeness.

The gains $\mathbf{K}_P$ and $\mathbf{K}_D$ for the NDI controllers were manually tuned by a trial-and-error method. A Monte Carlo approach could be used for this, but the trial-and-error method in this specific case proved to be more time-efficient, since running the simulations takes a long time, and manually selecting well-performing gains was more effective.

Finally, it should be noted that the INDI controller will be the controller of choice for the coming simulations. The focus of this study is not on control, but rather on the propulsion system and its impact on the controllability of the system. The INDI controller should be the most robust controller according to Chu (2018). This was tested for some of the later simulations, and no large difference between the different types were found. The INDI controller seemed to have a slightly more robust and smooth response to the inputs. The theory on the other controllers is still included in this chapter as it serves as the basis of the INDI controller, which uses the time-scale principle as well. The digital controllers, as implemented in the simulator, use a control time step of 0.04 s. The main simulation is run with a time step of 0.01 s.

6.4. ACTUATOR ASSIGNMENT

Appendix E contains the linearised EoM and the state-space representation of the vehicle. Throughout the flight, the controls can vary (e.g. aerodynamic surfaces at low altitudes, jets in space). To avoid having to change the equations and system representation each time the control vector changes, moment fractions are implemented in the model. These make the number of control inputs independent of the amount of actuators. This simplifies the controllers, and centralises the actuator assignment to a reusable block. Of course, a system for translating the moment fractions should be developed in order to make this approach work. Brinkman (2017) and Wu et al. (2000) have designed such frameworks for different applications. The algorithms developed by Brinkman (2017) are more suited to this application, so these are used in this study. Only a brief description of the algorithm is given in this report.

The actuator assignment algorithm works as follows. Based on the flight conditions, the maximum achievable control moments can be determined. These moments depend on the aerodynamic control surfaces, jets and the rocket engine. The roll, pitch and yaw jets deliver relatively constant moments, as they are practically independent from the flight conditions. The rocket engine can change its moment by varying the thrust angles and thrust force. The maximum moments delivered by the aerodynamic control surfaces are highly dependent on the flight conditions. They are extremely ineffective at

high altitudes, but become gradually more effective at closer to ground. Therefore, the actuator assignment includes an actuator selection, which dictates the used actuators at any flight condition. During ascent, since the effectiveness of the aerodynamic surfaces is so small, the full control is done by the rocket engine for $M > 5$ (Viavattene, 2018) or $\bar{q} > 5000$Pa.

The required commanded moments are obtained from the moment fractions as follows:

$$\mathscr{L}_{cmd} = \eta_x \mathscr{L}_{max} \tag{6.13a}$$
$$\mathscr{M}_{cmd} = \eta_Y \mathscr{M}_{max} \tag{6.13b}$$
$$\mathscr{N}_{cmd} = \eta_Z \mathscr{N}_{max} \tag{6.13c}$$

These commanded moments then need to be separated into the contributions of the aerodynamic surfaces and rocket engine. This is straightforward when only one of the options is used. When a combination is needed, the aerodynamic surfaces are always favoured over the propellant wasting engines. The engines are only activated when the aerodynamic controls cannot deliver the required moments. The reader is referred to Brinkman (2017) for a more thorough description of the algorithm. There are special routines for rudders and elevons, which are not described here, since aerodynamic surfaces are scarcely used during the hypersonic ascent phase.

7

SOFTWARE ARCHITECTURE & VERIFICATION

All the models and components described in the previous chapters should be integrated in one complete simulator, which can combine everything to create the required results. MATLAB Simulink provides a good platform to bring the models together, since it is specifically designed to simulate complicated systems. For the simulator to function appropriately, all components need to be connected in a logical and correct way, so the outputs of one model can flow into the next. This aspect of the simulator, namely its architecture, is described in Section 7.1. Afterwards, the numerical methods that are used in this simulator are elaborated on in Section 7.2. Finally, in Section 7.3, the models are verified and validated.

7.1. ARCHITECTURE

The simulator architecture can best be visualised in terms of blocks or components, each of which has its own functions. On a top level, the model can be represented by a few blocks, all containing a subset of blocks themselves. The blocks are connected through variables, which flow through the simulator from one block to the other. Some of the blocks change, adapt and use these variables to form their own. Dividing the software in modules provides a clear way of analysing the complete model, and it makes the model in its whole simpler to analyse an verify. Note that the skeleton of the simulator is based on the AOCS simulator by Mooij and Wijnands (2002). Many of the functionality is not changed, but a lot of components are added to this simulator. The digital controllers, as implemented in the simulator, use a control time step of 0.04 s. The main simulation is run with a time step of 0.01 s.

The software architecture of the simulator is presented in Fig. 7.1. Each of the blocks represents one branch of the simulator. These are the most important components of the simulator. Their functions are as follows:

- *Mission manager:* Here the commanded attitude profile is created. The commanded attitude consists of the three aerodynamic angles.

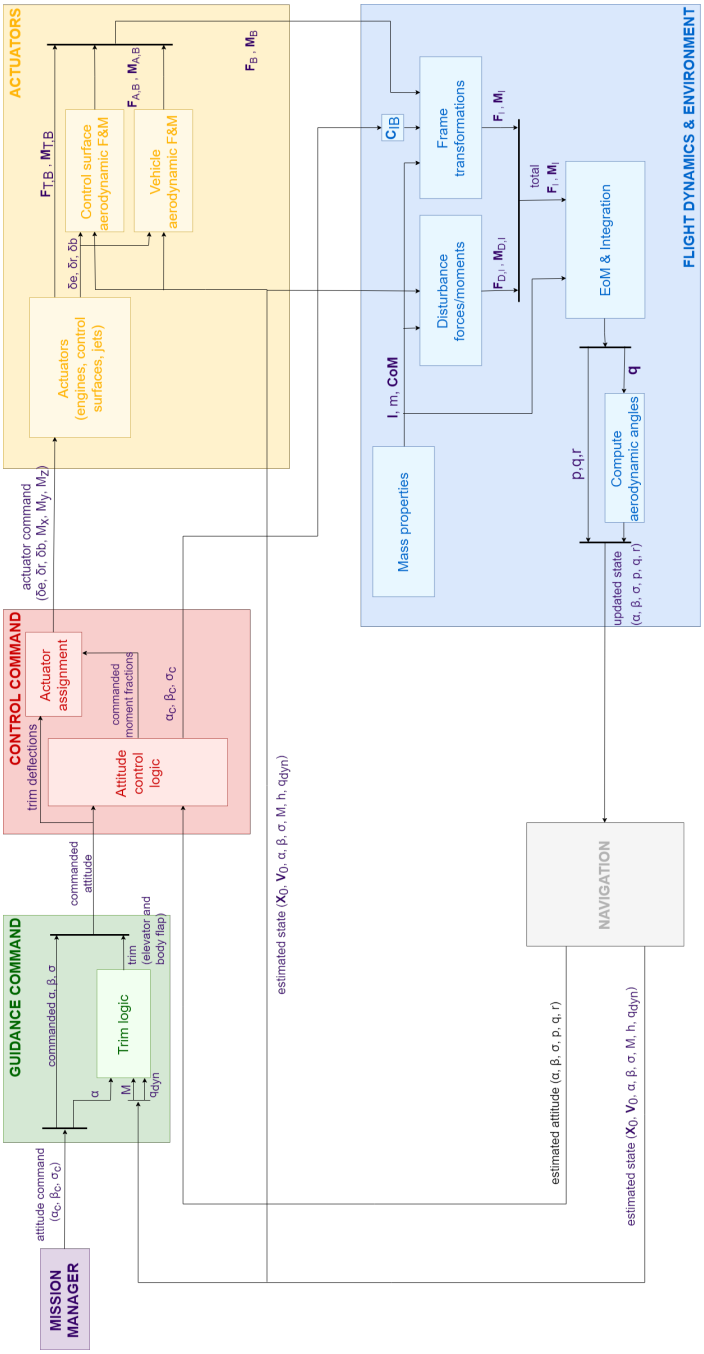


Figure 7.1: Software architecture. This image is an updated and reformed version of the architecture presented by Viavattene (2018).

- *Guidance command:* Guidance is not a major part of this study, since it is outside the scope. However, it is still important to include some trim logic to keep the vehicle stable during nominal flight. The trim logic computes the required elevon and body flap deflection, or thrust angles, to obtain trim stability.
- *Control command:* This block contains the attitude controller as well as the actuator assignment algorithm. Based on the commanded attitude, trim commands and estimated state of the navigation block, it will determine a suitable control command in terms of the different actuators present on the vehicle.
- *Actuators:* The control commands must be translated to control moments, which is done in this block. The actuators create wanted control moments, but also rather undesirable aerodynamic moments, which should be determined as well.
- *Flight dynamics & environment:* This block contains all the environmental models, flight dynamics and mass properties components of the simulator. Based on the previous state and the control (and other) moments, it will compute the new state of the vehicle at the current time step.
- *Navigation:* This component serves as the eyes of the system. Here, uncertainties can be added to the actual state to simulate how sensors work. No system can perfectly determine its own state, so control algorithms in reality have to deal with uncertainties in the input state. Navigation is outside the scope of this project, so the system is assumed to function perfectly, meaning the exact vehicle state is transmitted to the next time step.

The components that have changed from the original GAOCS simulator are:

- *Attitude control logic:* The NDI or LQR controller algorithms are added in this block. See Chapter 6 for a detailed description of the controllers.
- *Actuator assignment:* The actuator assignment block developed by Brinkman (2017) was updated to incorporate the rocket engines. During ascent at higher altitudes, aerodynamic surfaces are not effective, so these are replaced by the engines.
- *Actuators:* The actuator block, which translates the actuator command from the actuator assignment block to realistic deflections of the engines, aerodynamic control surfaces or thrusters, was extended by the propulsion model and other thrust-related functionality. Figure 7.2 shows the architecture of the thrust commanding block inside the actuator block. Engine properties and thrust input parameters serve as inputs to compute the thrust vector and location of the engines with respect to the COM of the vehicle as described in Chapter 3. These are then used to compute the TVC commands.
- *mass properties:* The mass properties block includes the inertia models described in Section 5.6.
- *Trim logic:* The trim logic block was slightly updated to add trim using the engines. During ascent, the engines are used for trim instead of the aerodynamic control surfaces.

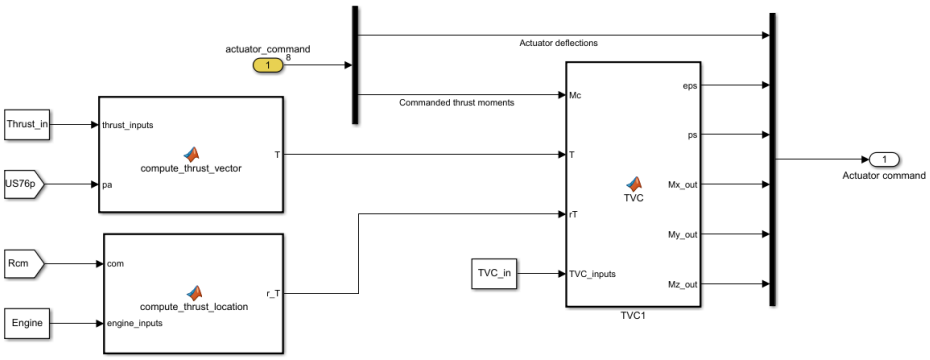


Figure 7.2: Software architecture of the thrust commanding block.

7.2. NUMERICAL METHODS

The simulations make use of several numerical methods to determine their outputs. Numerical methods are used for integration, differentiation, interpolation and much more. It is important to select appropriate methods too make the simulation run efficiently as well as with enough accuracy.

Firstly, numerical integration is discussed in Section 7.2.1. This is followed by interpolation techniques in Section 7.2.2.

7.2.1. NUMERICAL INTEGRATION

To start with, it is assumed that the differential equations to be solved are n-dimensional first-order equations of the form (Montenbruck and Gill, 2000):

$$\dot{\mathbf{x}} = \mathbf{f}(t, \mathbf{x}) \quad \mathbf{x}, \dot{\mathbf{x}}, \mathbf{f} \in \mathbb{R} \tag{7.1}$$

There are several methods for numerical integration, such as Euler integration, Taylor series approximations, Runge-Kutta (RK) methods, predictor-corrector methods and methods for extrapolation (Cohen, 2011; Hairer et al., 2008; Montenbruck and Gill, 2000). Each of these can be used to achieve accurate results, and have their own (dis)advantages. Euler integration is very simple, but lacks accuracy. Taylor series are inefficient as many derivatives need to be computed, which lead to additional errors. RK methods and extrapolation methods rely on function evaluations at different points. Overall, RK methods seem to be the favoured choice of integration scheme (Hairer et al., 2008). They achieve high accuracy and present high versatility.

There are many classes of RK schemes, ranging from the classical RK4 method, to more complex, but accurate, forms such as the RK8(7)-13 method. The theory behind RK integration schemes and its equations are not relevant to this study and the rader is referred to the book by Montenbruck and Gill (2000), which elaborates on the subject and includes many variants of the RK method.

One of the most popular RK methods is the embedded RK8(7)-13M (DOPRI8) method of Prince and Dormand (DOPRI). It can be used for a range of applications (Montenbruck and Gill, 2000) and provides sufficient accuracy as well as efficiency. The Dormand & Prince method is also the default method for the *ode45* ODE solver in MATLAB. It should be noted that *ode45* makes use of the DOPRI5 method, which is a 7-stage method of order 5 with an embedded method of order 4. Although it is less accurate than the DOPRI8 method, it still provides good results as shown in Montenbruck and Gill (2000). Because of these reasons, this method is used for the simulator.

7.2.2. INTERPOLATION

The reference data available usually does not come in the shape of continuous functions, but rather as a set of data points. Either one is lucky and all the required data points are available in the data set, or the more realistic case happens where data is needed between the given data points. The solution to this problem is interpolation.

The most basic interpolation scheme is linear interpolation. This method basically draws a straight line between two known data points, and assumes all output values are located on these lines. This method is not accurate, except when many data points are known without much distance between them. As this is mostly not the case for the reference data, more accurate methods should be analysed.

The book by Lindfield and Penny (2012) describes two basic methods of interpolation: Interpolation using polynomials and interpolation using splines. Both have similar performance in terms of accuracy, speed and robustness. Polynomials have the tendency to oscillate between data values, called Runge's phenomenon, which is not favourable. The spline method does not experience this drawback. Both of these methods are readily implemented in MATLAB and fairly easy to use. It is therefore not necessary to explain these methods in detail. As the spline method does not experience the unwanted oscillations, this method is used whenever interpolation is required.

Within the realm of spline interpolation, a distinction can be made between several spline interpolation methods (Erdogan, 2014; Hauser, 2009):

- **Linear splines:** This method is basically the same as linear interpolation, and thus not accurate enough for this application.
- **Quadratic splines:** A quadratic spline is a continuously differentiable, piecewise quadratic function, where quadratic includes all linear combinations of the basic monomials. Quadratic splines are not used in applications as much as cubic splines (Erdogan, 2014).
- **Cubic splines:** Cubic splines are often preferred in applications where smooth interpolation is crucial, such as this one.
- **(Cubic) Hermite splines:** This method is a variation on the cubic spline method. Here, the original technique is modified by combining cubic splines with a set of other polynomials, called the cubic Hermite polynomials. As opposed to the other methods, this one provides not only continuity of the first order derivatives of the curve, but is also able to match the set of chosen slopes. The Hermite form thus has two control points and two control tangents for each set of points.

Choosing the appropriate method depends on the problem. The linear method is most suitable for linear data as it is simple and will not overshoot. Alas, the databases used for this study are not linear. According to Erdogan (2014), quadratic splines oscillate and overshoot the data, which makes this approach not suitable to acquire any relevant data. Cubic splines give better results and do not show large oscillations. Cubic Hermite splines overall show the best behaviour, there is no overshooting and gives smooth results. The Hermite splines are most likely to give the required results without unwanted oscillations, which is why this method is chosen. The reader is referred to Erdogan (2014) for more information on this topic.

7.3. VERIFICATION

Without thorough verification of the developed models and software, it is impossible to make any sound conclusions or present the results. The verification procedure is necessary to prove that the models can reproduce the same results as the theoretical models. Verification is important at each step in the software development, ranging from checking the smallest singular functions (unit tests), to the larger blocks of functions (system tests), to the fully integrated system (integration tests). Verification is done both intuitively, by checking the output of certain functions and comparing this to the expectations, and by creating dedicated tests with reference data.

All the blocks in the architecture described in the previous section should be verified. Even though many of these blocks were not specially developed for this study, and taken from other developers, it is necessary to do an acceptance test for all models used, even if it should already have been verified. For example, most of the flight dynamics were taken from the generic AOCS simulator from Mooij and Ellenbroek (2011); Mooij and Wijnands (2002), which was verified in their reports. It is therefore not necessary to verify every piece of their software, but a general acceptance test should and will be carried out. The verification of all models that were not specifically developed for this study is summarised in Table 7.5.

7.3.1. CHEMICAL EQUILIBRIUM COMPOSITION & ROCKET PERFORMANCE

The chemical equilibrium composition and rocket performance algorithms from Chapter 3 can easily be verified by comparing the outputs to the results of the Rocket Propulsion Analysis software. Ponomarenko (2009) describes several test cases, which are used here to verify the model. The main input parameters of the test cases are summarised in Table 7.1.

Tables 7.2 and 7.3 give the resulting values from the model, and their counterparts from the RPA software. Clearly, results are similar to the RPA results. The small differences can be explained due to the simplifying assumptions, and the integration scheme used, which might be different from the RPA models. However, this also confirms that the assumptions listed in Sections 3.1.1 and 3.2.1 are valid assumptions, which do not have a detrimental impact on the results. It should be noted that the specific impulse in the table below is expressed in meter per second, insted of the usual meter. This is because RPA uses this expression for the specific impulse, whereas this value is usually divided by the gravitational acceleration at sea-level.

Table 7.1: Verification test cases (Ponomarenko, 2009).

Parameter	Test case 1	Test case 2	Test case 3	Unit
Oxidiser/Temperature	LO_2/90.17	LO_2/90.17	LO_2/90.17	K
Fuel/Temperature	LH_2/20.27	$CH_4(L)$/111.64	RP-1/298.15	K
OF ratio	5.5	3.2	2.6	-
Chamber pressure	10	10	10	MPa
Nozzle exit area ratio	70	70	70	-

Table 7.2: Rocket performance parameters verification.

	Test case 1		Test case 2		Test case 3		
Parameter	**Model**	**RPA**	**Model**	**RPA**	**Model**	**RPA**	**Unit**
T_{cc}	3432.21	3432.01	3565.80	3566.07	3723.22	3723.63	°
c^*	2345.51	2345.30	1861.10	1862.41	1800.53	1800.60	ms^{-1}
C_T	1.87	1.87	1.91	1.91	1.92	1.92	-
I_{sp}	4549.14	4549.28	3701.09	3701.07	3596.61	3596.50	ms^{-1}
M_e	4.69	4.69	4.44	4.44	4.39	4.39	-

Table 7.3: Verification of the mole fractions in the combustion chamber.

	Test case 1		Test case 2		Test case 3	
Species	**Model**	**RPA**	**Model**	**RPA**	**Model**	**RPA**
CO	NA	NA	0.19732	0.19733	0.31521	0.31521
CO_2	NA	NA	0.11702	0.11704	0.15383	0.15383
$COOH$	NA	NA	0.00001	0.00002	0.00003	0.00003
H	0.02774	0.02775	0.09899	0.09898	0.02686	0.02686
H_2	0.30155	0.30152	0.09899	0.09898	0.07953	0.07954
H_2O	0.64016	0.64016	0.49168	0.49169	0.33333	0.33329
H_2O_2	0.00001	0.00001	0.00001	0.00002	0.00002	0.00002
HCO	NA	NA	0.00014	0.00002	0.00001	0.00004
HO_2	0.00001	0.00001	0.00009	0.00009	0.00012	0.00011
O	0.00141	0.00140	0.00641	0.00641	0.01119	0.01119
O_2	0.00116	0.00115	0.01220	0.01221	0.01788	0.01785
OH	0.02799	0.02799	0.05462	0.05463	0.06203	0.062023

7.3.2. TANK INERTIA

Most of the inertia model can be verified by analysing the figures from Section 3.3. Figure 3.9 shows the MMOI around the centre of the tank and around its own COM for a horizontal cylinder without inclination. The marks indicate several points for which analytical equations are known, e.g. half-full cylinder and full cylinder. The marks coincide with the lines, which verifies this part of the model.

The fill case selection is verified in Fig. 3.14. Case A can only be active when there is still more than half the full mass of propellant in the tank, whereas case D is the reverse. This is shown in the figure by the vertical dotted line. Furthermore, case B can only be

active when the inclination is smaller than the limiting inclination. The opposite is true for case D. These characteristics are again clearly shown in the Figure by the horizontal dotted line.

Finally, several limit cases were tested to verify the inertia and centre of mass calculations for the inclined tanks. Looking at Fig. 3.4, there are multiple limit cases, which can be distinguished:

- **Limit case 1:** case A ($h_A = 2R, l_A = L$) = case B ($h_{B,1} = 2R, h_{B,2} = 0$) = case C ($l_{C,1} = 0, l_{C,2} = L$) = case D ($h_D = 2R, l_D = L$) → Half cylinder cut through diagonal ($\alpha = \alpha_L$)
- **Limit case 2:** case A ($h_A = 2R, l_A = L$) = case B ($h_{B,1} = 2R, h_{B,2} = 2R$) = case C ($l_{C,1} = L, l_{C,2} = L$) → Full cylinder ($\alpha = 0$)
- **Limit case 3:** case B ($h_{B,1} = 0, h_{B,2} = 0$) = case C ($l_{C,1} = 0, l_{C,2} = 0$) = case D ($h_D = 0, l_D = 0$) → Empty cylinder ($\alpha = \alpha_L$)
- **Limit case 4:** case B ($h_{B,1} = R, h_{B,2} = R$) = case C ($l_{C,1} = L/2, l_{C,2} = L/2$) → Half horizontal cut cylinder ($\alpha = 0$) and half vertical cut cylinder ($\alpha = \pi/2$)

For all above cases, the resulting MMOI and COM were compared to the analytically computed values. All of the results were as expected.

7.3.3. VEHICLE INERTIA

The vehicle inertia model comprises two components: the determination of the amount of left-over mass in each tank, and the calculation of the COM and MMOI of the vehicle including the variable propellant mass. Both of these components are verified in this section.

LEFT-OVER PROPELLANT MASS

The propellant mass in each tank of the vehicle can be determined using the algorithm described in Section 5.6. One example of the output of this algorithm was already shown in Fig. 5.2. This figure clearly shows how the algorithm works, and verifies one situation, namely where all tanks are depleted at the same time. The dashed lines indicate the total mass in each tank. All the tanks stop depleting when they reach their maximum mass, as it is impossible to take more mass out of a tank than it has.

To verify the algorithm, four cases are tested. Figure 7.3 shows the results. Each of the cases tests a different aspect of the input sequence, which is the main variable for this algorithm that could induce errors. The upper left graph shows the case where all the oxidator and fuel tanks are emptied sequentially, i.e. never two tanks of the same type depleted concurrently. The plot shows this clearly, as one line only starts to rise when the previous flattens. In the second case (upper right), the order of depletion is scrambled, showing similar results to case 1. The same is true for the third case (bottom left).
Finally, the fourth case includes tanks being depleted at the same time (tank 2 and 3, and tank 4 and 5). The results are again as expected. More cases were tested, e.g. with more or less than five tanks, different sizes and multiple sequence combinations. However, not all these results are included in the report as most of them are redundant and do not give extra information.

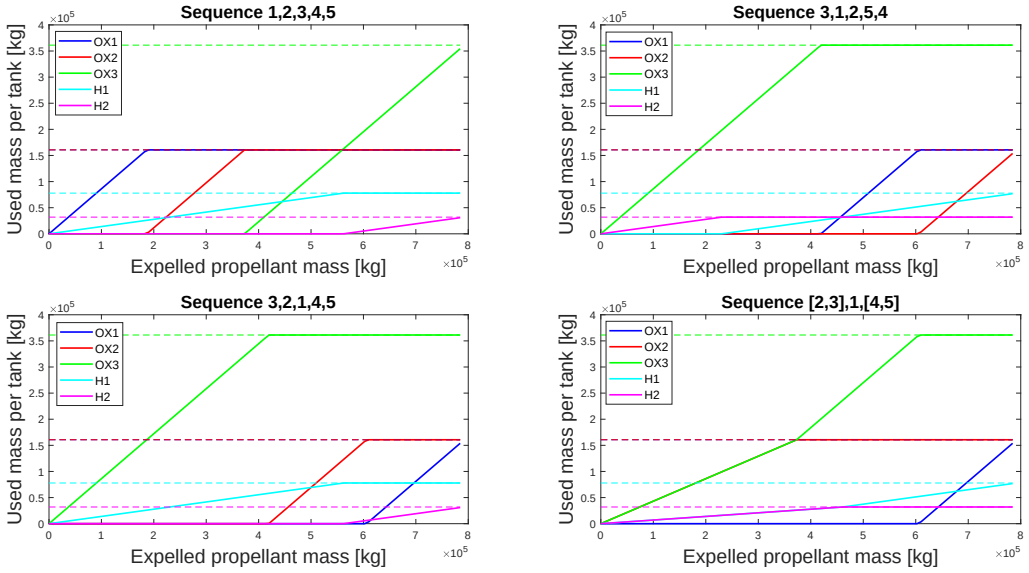


Figure 7.3: Verification of the *get_used_masses* algorithm. Used mass per tank with respect to total expelled mass for a vehicle with five tanks for different depletion sequences. The tank dimensions are taken from the FSSC-1 vehicle (Daimler-Benz Aerospace, 1996c).

COM AND MMOI COMPUTATION

Since it is difficult to find reliable and accurate inertia data for actual vehicles, a different acceptance test was developed for the inertia calculations. The inertia algorithm is applied to a uniform cube with a certain mass, inertia, etc., with several standard tanks added in different locations. In this way, the results can more easily be interpreted, as the shapes in this test are familiar. For this case, the solid cube is the dry mass. The actual size of the cube or tanks is not of much importance for this example, so for simplicity, a cube with a side length d of 10 m and mass m_{cube} of 10,000 kg is used. The COM of the cube is in its centre and its inertia matrix is given as:

$$\mathbf{I}_{cube} = \text{eye}(3,3)\frac{1}{6}m_{cube}d^2 \tag{7.2}$$

The inertia of the tanks can be computed using the model from Section 3.3, and the total inertia is calculated as detailed in Section 5.6. To verify the models, the inertia of several cases is computed for ranges in inclination angle as well as used mass. For each of the cases, the number of tanks, their size and location is changed, to see the influence of all parameters. The results for one of the cases are presented in Figs. 7.4 and 7.5. Due to the size of the resulting figures, not all test cases are included in the report.

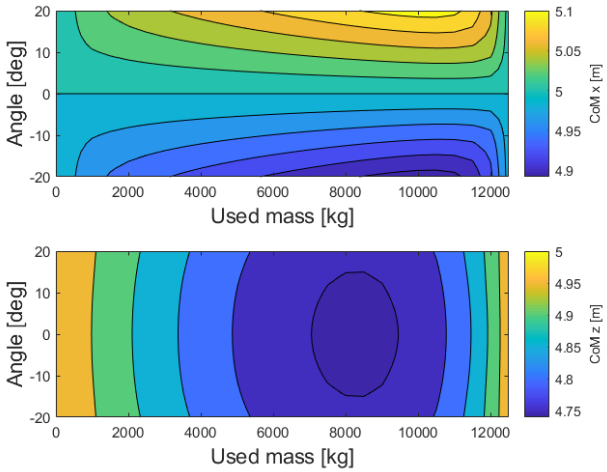


Figure 7.4: Verification of COM computation for a solid cube with a mass of 5000 kg and side of 10 m, with two tanks of length 2 m, radius 1 m and propellant density of 1000 kg m^{-3} at its centre of mass.

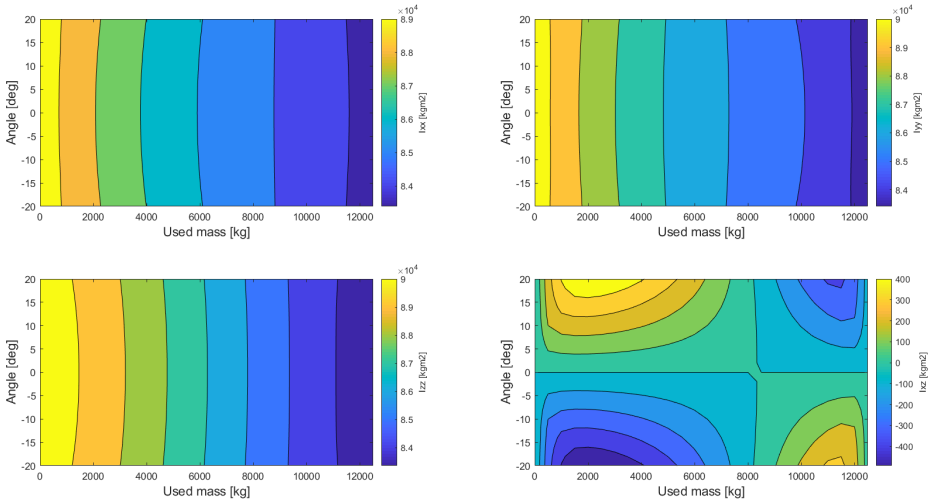


Figure 7.5: Verification of inertia computation for a solid cube with a mass of 5000 kg and side of 10 m, with two tanks of length 2 m, radius 1 m and propellant density of 1000 kg m^{-3} at its centre of mass.

The first example, the solid cube and two tanks located at the centre of mass of the cube, shows expected results. Two tanks located at the same point is not realistic, but as verification, it fits the purpose. Overall, the COM and inertia do not change much, but there is a visible shift. The y-axis COM position is not included as it does not change due to the symmetry of the test case. For zero inclination, the x position of the COM remains

constant, as there is symmetry along the yz-plane. Only when the angle changes, the CoM_x shifts accordingly, either to the right or left, depending where the larger part of the propellant is located. The COM_z position generally shifts downwards, as the propellant is always located at the bottom of the tank. Interestingly, when the remaining mass in the tank sinks below a certain degree, the COM rises back towards its original position in the centre of the cube. This is also logical. At this point, the mass in the tank is so low that it does not influence the COM position significantly.

When the tanks are empty, the COM and inertia become equal to the one of the solid cube. The inertia values show a consistent decrease when mass is expelled. Less mass generally means smaller inertia. Since the tanks are located at the centre of the cube, their influence on the inertia through Steiner's theorem is relatively small, which is evident from the minor change with angle of attack and overall with decreasing propellant mass. In the following cases, the shift in inertia will be much larger when the tanks are located further from the centre of the cube.

7.3.4. CONTROL SYSTEM

The verification of the control system has two main component. First of all, the LQR controller as described in Section D needs to be verified. Afterwards, this controller can be used to verify the other controllers.

NDI CONTROLLER

The response of the system using the NDI controller is included in Fig. 7.6. The simulations were carried out with many different gains to find a suitable gain matrix. The NDI controller, in this case, gives a faster response to the input, without having a larger overshoot or steady-state error.

Fig. 7.7 gives a similar view on the response of the NDI controller to a step input. The simulations were done for four points in the re-entry trajectory of HORUS, at $M = 5, 10, 15, 20$. Although the conditions are very different, the response looks similar for all of the simulations. Only at $M = 15$, there is a small deviation in the pitch rate. However, the pitch rate never increases beyond $0.7\,^{\circ}\mathrm{s}^{-1}$, which is within the set control limits. The reason that the re-entry trajectory is chosen for these simulations is that their are much more dependencies on other systems during the ascent trajectory. For example, during ascent, the inertia changes, the propulsion system is active, sloshing has a larger influence, etc.

Table 7.4: Input parameters for the controller verification. These variables are based on the input parameters for the bank reversal manoeuvre of HORUS Brinkman (2017).

Parameter	Value	Unit
Altitude	66209.4	m
Dynamic pressure	2297.6	Pa
Mach number	18.78	-
Flight-path angle	0.64	°
Angle of attack	40	°
Bank angle	59.8	°

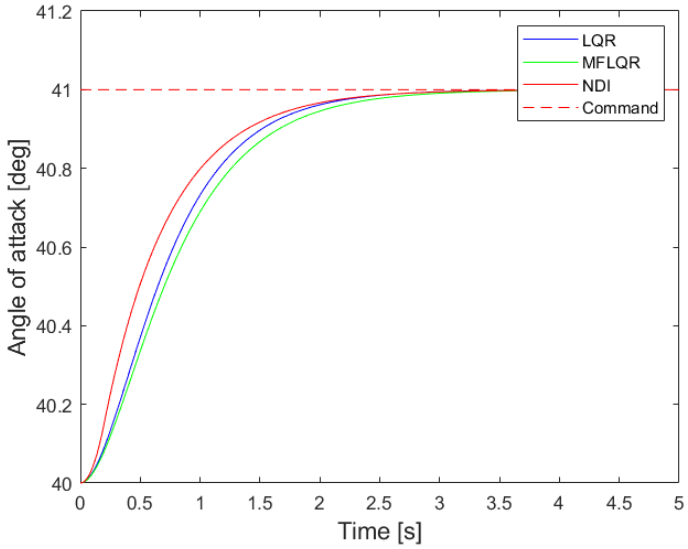


Figure 7.6: Comparison of the LQR, MFLQR and NDI controllers for a step input in angle of attack at flight conditions given in Table 7.4.

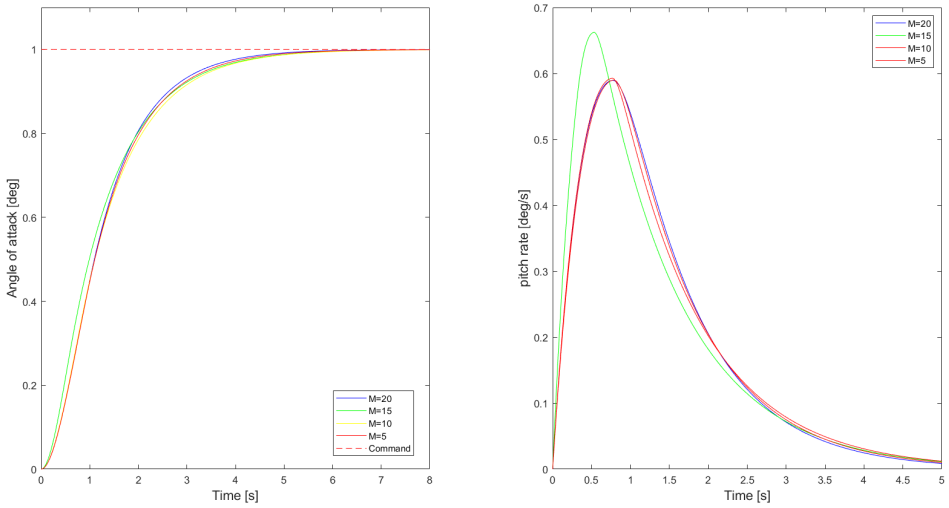


Figure 7.7: Comparison of a 1-degree step input in angle of attack at four different points in the re-entry trajectory of Horus using the NDI controller.

7.3.5. SLOSHING

The sloshing model was verified in two ways. First of all, the sloshing parameters were compared to results from several references. The plots from Fig. 3.19 present several of the parameters. These graphs were compared to the results by Raouf (2006) and Basurto et al. (1966). The values were found to be equal. Then, the equations of motion were tested in several controlled experiments. The response to several general inputs (e.g. $F_x = 1\,\mathrm{N}$ or $M_y = 1\,\mathrm{N\,m}$) was examined to see whether it shows expected behaviour. As an example, for a positive force in x-direction, the initial response of the tank should be to move forward, whereas the sloshing masses should move backwards w.r.t. the moving container. Similar tests were done for the rotational motion.

7.3.6. OTHER MODELS

Table 7.5 summarises the verification of other important components of the simulator that were not specifically developed for this study.

Table 7.5: Verification of all models or components not especially developed for this study.

Component	Test description	Verified?
US76 Standard Atmosphere	Verified by MATLAB developers/users and compared results to other data sources	Yes
Flight dynamics	Verified by Mooij and Wijnands (2002) and Mooij and Ellenbroek (2011), and several tests done to confirm equations of motion	yes
(MF)LQR controller	Results of Viavattene (2018) and Brinkman (2017) recreated with equal inputs	yes
Actuator assignment	Tests of Brinkman (2017) repeated with equal and different inputs	yes

8

PROPULSION & INERTIA MODEL SENSITIVITIES

This chapter describes the sensitivity of the propulsion and inertia models. It is important to examine how different parameters influence the resulting thrust and inertia values. In this way, it is possible to get an idea of the sensitivity of these variables, as well as investigate what parameters are important in the models. high sensitivity is generally not favourable, as it indicates that small changes in flight conditions can have a serious impact on the system. It also relates to the stability of the system. Imagine an object at the bottom of a valley. This is a stable system and cannot easily be influenced by a push or force. On the other hand, an object at the summit of a hill can easily be moved by a small force, which indicates this system is much more sensitive to force, and therefore unstable.

A second reason for this investigation is to determine if certain parts of the models can be taken out of the simulation and be kept constant over one simulation period. The simulations are time-consuming, so leaving out certain parts that do not contribute over a small time, is definitely beneficial, certainly when Monte Carlo analyses are done. As an example: if thrust would remain practically constant over a small period of time, it would be useful to take the whole propulsion model, as described in Chapter 3, out of the simulation. Of course, this can only be done if sensitivity of this model is low, hence the need for this study.

Firstly, in Section 8.1, the propulsion model is analysed to investigate the sensitivity of the thrust and other model parameters. This measure of the thrust sensitivity is important for the later simulations, as a change in thrust can have large consequences for the stability and control of the vehicle. Next, the inertia model of the vehicle is studied. Here, it is important to identify the extent to which the inertia can change over the course of a short simulation.

8.1. PROPULSION MODEL SENSITIVITIES

Before the full vehicle model is analysed, the sensitivities in the propulsion model alone are investigated. This is done because the propulsion model of the rocket engine does not depend on many external parameters. Rocket engines basically function in the same way no matter the external conditions. The only external parameters for the computa-

Table 8.1: Input and output parameters for the propulsion model sensitivity analysis.

Parameter	Symbol	Unit
INPUT		
Oxygen to fuel ratio	OF	-
Nozzle exit area	A_e	m^2
Nozzle throat area	A_{th}	m^2
Nozzle area ratio	$A_{ratio} = \frac{A_e}{A_{th}}$	-
Combustion chamber pressure	p_c	Pa
Ambient pressure	p_a	Pa
OUTPUT		
Thrust	T	N
Mass flow	$\dot{m}$	$kg\,s^{-1}$
Thrust coefficient	C_T	-
Specific impulse	I_{sp}	s
Nozzle exit Mach number	M_e	-
Exit velocity	u_e	$m\,s^{-1}$
Exit pressure	p_e	Pa
Mole fractions of combustion products	x_j	$kg\,mol\,kg^{-1}$

tion of the thrust is the ambient pressure. This variable can easily be determined using the US76 standard atmosphere given in Section 4.2. The other involved variables are either inherent parameters of the engine itself, such as its dimensions, combustion chamber pressure, etc., or depend on the chosen propellant and propellant tanks.

The choice of propellant has many implications for the design of the vehicle. Each propellant combination has its own advantage, such as higher specific impulse, lower mass, less volume required, more safety or easier to store in the tanks. Because of this, changing the propellant does not only influence the thrust, but also the mass of the vehicle, which leads to a snowball effect, changing the complete vehicle. Since a complete redesign of the vehicle is outside the scope of this study, the changes of propellant are only investigated on a high-level basis, by looking at the direct influence on the thrust and related parameters, without resorting to a more in-depth redesign of the vehicle. The propellant tanks are assumed to only relate to the rocket engine through the mass flow.

The actual shape and location do not have an impact on the resulting thrust, as long as the mass flow remains constant. Therefore, the mass flow is taken as one of the main research parameters, without resorting to other details about the tanks. Remember that the mass flow itself also depends on the combustion parameters, such as the density and exhaust velocity of the flow, and the nozzle exit area according to Eq. (3.25). In later sections, the impact of the tank locations and dimensions on the inertia, and motion of the vehicle is investigated in more detail.

The main goal of this section is to research how different variables impact the thrust, and other important propulsion model parameters. The most important in- and output parameters are given in Table 8.1. In the end, the resulting thrust is the most important

parameter, since it is the only variable that interacts with the vehicle model. However, the other parameters are included to get a better idea of the model sensitivities and inaccuracies. The sensitivity of an output variable with respect to an input parameter is obtained by a grid search technique. The output variables are computed for a range of the input parameter, where the range depends on the nominal value of the parameter. The results can be presented in the form of figures or tables, but the tables are chosen mainly to save space in the report. As a second form of analysis, a Monte Carlo (MC) analysis is done on the set of input parameters, to define the worst-case and best scenarios that can occur. The MC analysis is chosen for this as the run time of the grid search technique scales exponentially with the number of dimensions, whereas the MC method represents a simplified way of analysing different dimensions (a.k.a. different parameters) at the same time.

The propulsion model is applicable to numerous rocket engines. Therefore, to adequately study the sensitivity of this model, it is important to study multiple cases, and not only the one case of the HORUS vehicle. Furthermore, the reference data on the rocket engines of HORUS is difficult to find, and not constant through different sources, which decreases the reliability of the reference data. Therefore, the first case that is analysed is the upper stage of the FSSC-12 vehicle (Daimler-Benz Aerospace, 1996a). This vehicle uses a $LO_2 + LH_2$ bi-propellant combination. The second analysed case is the FSSC-1 rocket engine (Daimler-Benz Aerospace, 1996c). Here, the same propellants are used, but the input parameters are different. Then, to analyse different propellants, the cases stated in Table 7.1 are investigated, as these are relevant cases for which the exact parameters are known, and verified. Through these five cases, it should be possible to derive several conclusions regarding the sensitivity of the propulsion model.

8.1.1. FSSC-12 UPPER STAGE

The nominal inputs and outputs for the FSSC-12 upper stage are given in Table 8.2. Table 8.3 presents the sensitivity results. The table shows the minimum and maximum deviation of the nominal results for an input parameter range of 10%. This means that the input parameter is ranged from 90% to 110% of its nominal value. The cells in red show the results where there is more than 10% deviation from the norm. These are the most important results, as they show what output variables are influenced by the input parameters the most. For example, the exit pressure deviates by more than 15% for a maximum 10% change in oxygen to fuel ratio. This indicates that the exit pressure has a high dependency on OF, which is not a favourable condition. It means that a slight change in OF could have a quite large impact on the exit pressure. High sensitivities are never desired, as they can make a model unstable.

The exit pressure deviation is still within bounds, however this can not be said for the mole fractions. The oxygen mole fraction more than doubles for a 10% change in OF. This indicates that the oxygen to fuel ratio is arguably the most important input parameter for the combustion process. Nonetheless, this parameter does not have as much of an influence on the resulting thrust, only around ±2%. This demonstrates that large deviations in the combustion gas constituents do not have a major impact on the resulting thrust or other important rocket parameters. Furthermore, it is also visible that the exit pressure is the most dependent output, as it experiences a high deviation from multiple

inputs. Finally, as can be expected from the thrust equation:

$$T = \dot{m}u_e + A_e(p_e - p_a) \tag{8.1}$$

the mass flow has a quasi-linear relation with thrust. The parameters that influence $\dot{m}$ therefore also impact the thrust. Good examples are p_c and A_e, both of which result in a thrust deviation of about ±10%. The exit area has a one-to-one relation with mass flow, indicated by the exact ±10% sensitivity result. This is expected from the mass flow equation. The two area cases show approximately the same results for all parameters except the thrust and mass flow, which are influenced more by a the exit area.

The large deviations of the mole fractions do not necessarily mean the model is unstable. The nominal mole fractions given in Table 8.2 have extremely small values, except for H_2O. A small inaccuracy in the computations can therefore have a relatively large impact on the results. The most reliable parameter to look at is the one for H_2O. Here, there is only a deviation of a few percent, which does suggest the model is still stable.
Even though these parameters have large deviations, their eventual impact on the resulting thrust is relatively small.

Next, a MC simulation with 10000 runs is carried out. Here, all the input parameters are randomly chosen within their respective ±10% ranges (e.g. $OF_{nom} = 7.2 \rightarrow OF_{range} = 6.48 \Rightarrow 7.92$). The variables are randomly chosen using the MATLAB *rand* function, which returns uniformly distributed numbers. The results are presented in the form of histograms in Figs. 8.1 to 8.3, as these provide a clear and convenient way of demonstrating the sensitivity of the model parameters.

Table 8.2: Nominal input and output parameters for FSSC-12 rocket engine sensitivity analysis (Daimler-Benz Aerospace, 1996a). Reference altitude for ambient pressure is 15 km.

Parameter	Value	Unit
	INPUT	
OF	7.2	-
A_e	4.4488	m^2
$A_t h$	0.01618	m^2
A_{ratio}	275	-
p_c	244.5	bar
p_a	12111	Pa
	OUTPUT	
T	769.50	kN
$\dot{m}$	176.78	$kg s^{-1}$
C_T	2.04	-
I_{sp}	447.76	s
M_e	5.37	-
u_e	4568.44	$m s^{-1}$
p_e	5061.60	Pa
$x_{c,j}^{(1)}$	$[0.8677, 0.0267, 0.01862, 0.07820, 0.0067, 0.0016, 0.0004, 0.0001]$	$kg mol kg^{-1}$

(1) $j = [H_2O, O_2, H_2, OH, O, H, HO_2, H_2O_2]$

Table 8.3: Propulsion system sensitivities in percentages for propellant combination LO_2 + LH_2 for a range of 10% around the nominal input parameters. Reference vehicle is the FSSC-12 upper stage. Nominal conditions given in Table 8.2. The red cells indicate sensitivities larger than 10%.

	OF		p_c		A_e		A_{th}		p_a	
T	-2.04	1.70	-10.62	10.62	-10.42	10.39	-0.42	0.38	-0.69	0.69
$\dot{m}$	-2.39	2.65	-9.91	9.90	-10.00	10.00	0	0	0	0
C_T	-1.62	1.20	-0.060	0.066	-0.35	0.37	-0.38	0.34	0	0
I_{sp}	-0.92	0.36	-0.79	0.65	-0.47	0.35	-0.42	0.38	-0.69	0.69
M_e	-5.48	4.78	-0.13	0.12	-1.45	1.62	-1.60	1.47	0	0
u_e	-1.42	0.79	-0.031	0.028	-0.34	0.37	-0.38	0.34	0	0
p_e	-12.36	15.70	-9.69	9.66	-12.71	13.08	-11.57	14.55	0	0
T_{cc}	-1.65	0.35	-0.37	0.33	0	0	0	0	0	0
x_{c,H_2O}	-4.34	3.42	-0.25	0.23	0	0	0	0	0	0
x_{c,O_2}	-61.44	103.27	-2.47	2.75	0	0	0	0	0	0
x_{c,H_2}	-32.99	50.50	-0.67	0.74	0	0	0	0	0	0
$x_{c,OH}$	-25.15	17.12	-1.23	1.34	0	0	0	0	0	0
$x_{c,O}$	-44.05	42.47	-3.29	3.70	0	0	0	0	0	0
$x_{c,H}$	-18.42	12.11	-2.67	3.00	0	0	0	0	0	0
x_{c,HO_2}	-54.57	71.40	-2.36	2.13	0	0	0	0	0	0
x_{c,H_2O_2}	-41.67	42.07	-5.44	5.15	0	0	0	0	0	0

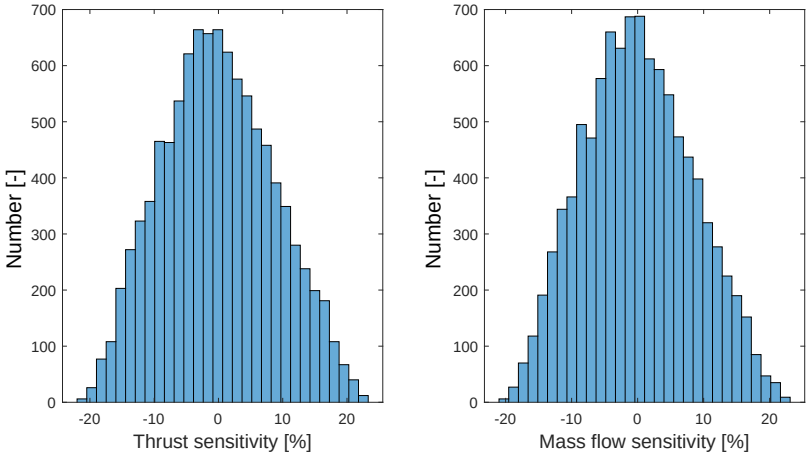


Figure 8.1: Thrust sensitivity resulting from a Monte Carlo simulation with 10000 runs, for an input variation of 10% for the FSSC-12 rocket engine.

Fig. 8.1 shows that the maximum thrust and mass flow sensitivity ranges around 20%. Since these parameters are influenced by all the input variables, this can be expected. Despite the large deviations, the thrust remains close to its nominal value. In Fig. 8.2, the main rocket parameters sensitivities are shown. Most of these do not show a large value, except for the exit pressure, which was already established in the table as well. The exit pressure is highly dependent on several factors, such as the shape of the nozzle and the throat conditions. For most of the parameters, the histograms seem to be centred around zero percent, which is the most favourable condition. Overall, the sensitivities lean more

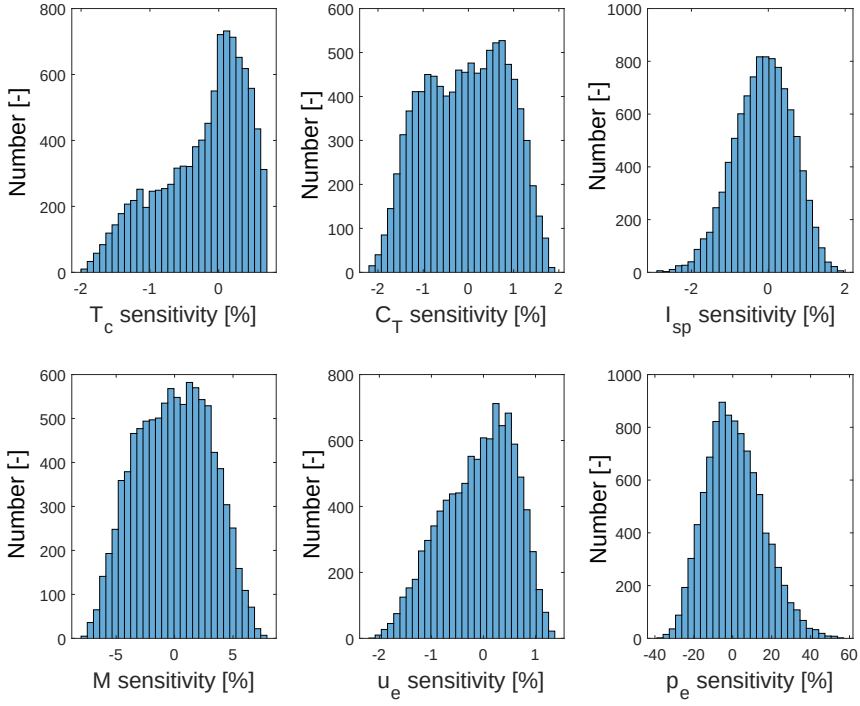


Figure 8.2: Rocket parameters sensitivity resulting from a Monte Carlo simulation with 10000 runs, for an input variation of 10% for the FSSC-12 rocket engine.

towards the negative, see for example the combustion temperature (T_c) sensitivity plot. This is because the analysed nominal values are already optimised for the mission, which means on average, most other combinations will result in lower outputs.

The intermediate combustion outputs in the form of the mole fractions are presented in Fig. 8.3. Again, the mole fractions prove to be very volatile with respect to changing input variables. As explained before, x_{c,H_2O} is the most important fraction, since most of the combustion gas is made up of H_2O. This fraction has a far lower sensitivity, ranging from -4%to2%. As long as the main constituents of the gases do not have a large sensitivity, the model remains stable. When investigating different propellant combinations, this feature should be present again.

One could note that using a 10% range is quite arbitrary. The same tests were performed for other input ranges: 5%, 15%, 20%, 25% and 30%. This to determine if these results are of a linear nature. The results for the thrust sensitivity are shown in Fig. 8.4. First of all, the results for almost all parameters show linear behaviour. So a change in input deviations scales linearly with the output sensitivity. This is true for all variables, except the OF ratio, where the maximum sensitivity stops at around 1.7%. The exact origin of this feature is not exactly known. One possibility could be that the optimal OF

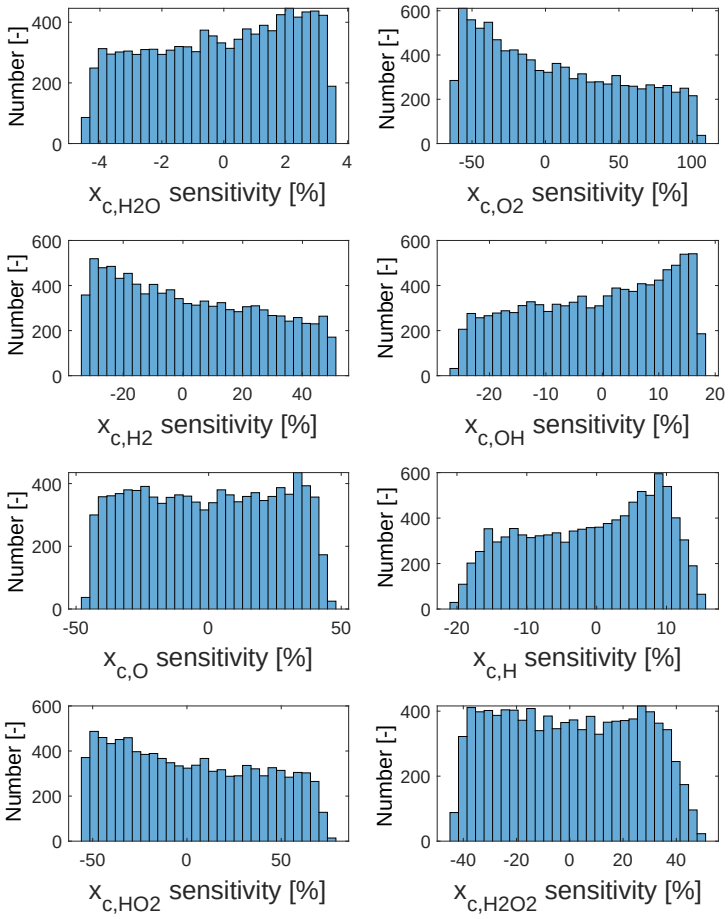


Figure 8.3: Combustion chamber mole fraction sensitivity resulting from a Monte Carlo simulation with 10000 runs, for an input variation of 10% for the FSSC-12 rocket engine.

ratio is reached at some point, at around 10% deviation, after which an increase in OF does not have a positive effect on the thrust. Since the results are overall linear, the exact value of the used range is not of utmost importance. Therefore, the value of 10% is kept for the following tests.

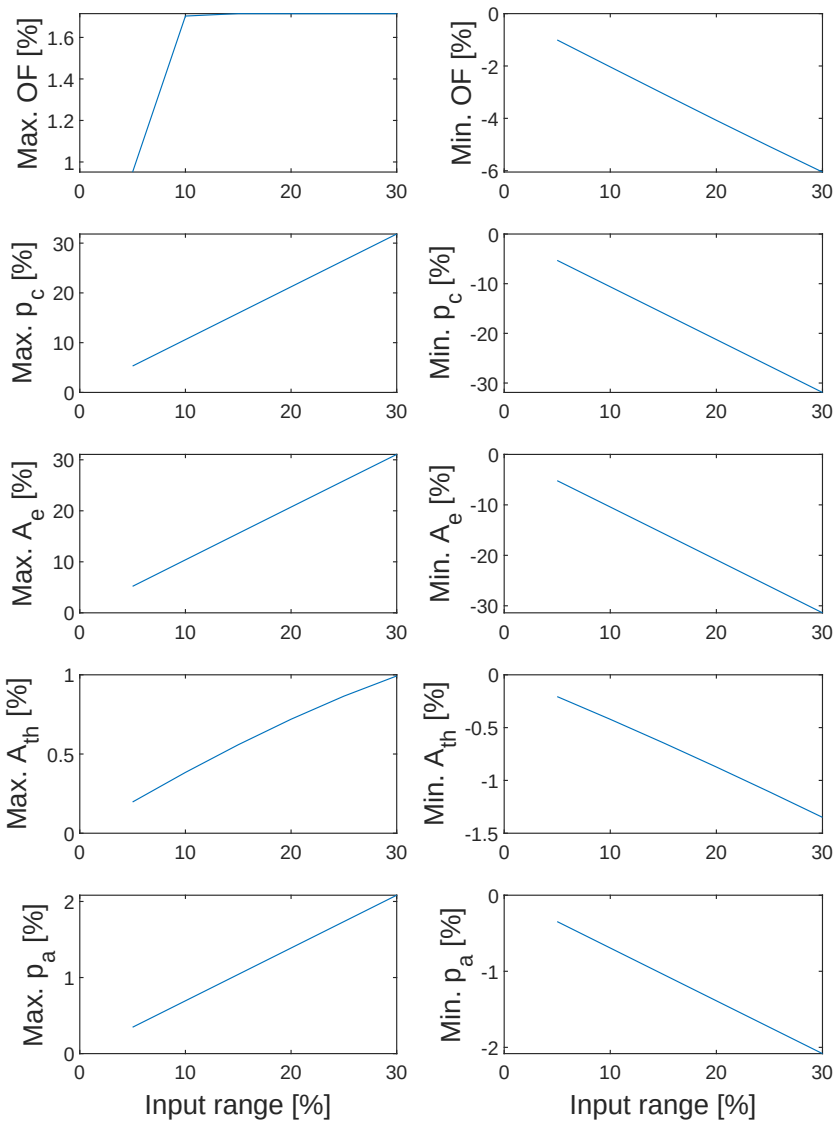


Figure 8.4: Thrust sensitivity for different input ranges of different input variables. All graphs represent thrust sensitivity. The y-axis labels represent which input variable was used (e.g. the upper left figure shows the maximum sensitivity values from the MC simulations for deviations in OF ratio).

8.1.2. FSSC-1

The nominal inputs and outputs for the FSSC-1 vehicle are stated in Table 8.4. The engine dimensions are quite different than the previous example, which makes it an interesting second case to analyse. The sensitivity results are given in Table 8.5.

The results are quite similar to the ones for FSSC-12. The *OF* ratio is still the most 'influential' parameter for the combustion model outputs, whereas the exit pressure shows the largest sensitivity, certainly with respect to changing areas. The thrust again has a high dependency on the chamber pressure and exit area. There also seems to be a trend towards the negative. This feature was already explained in the previous section. From these results, it can be concluded that the model is versatile and works properly and similarly for different cases.

The results of the MC simulations are presented in Figs. 8.5 and 8.6. The mole fractions are not included for this example since the results are very similar to the ones from the previous example, and these do not have a large influence on the more important parameters. Generally, the thrust sensitivity is similar to the previous case. The mass flow and thrust have a linear relationship. The deviation in the negative direction is larger again, which is also due to the optimisation of the nominal case. On average, changing the optimal values has a detrimental effect on the thrust. The reason that there are still input combinations that result in higher thrust or specific impulse values is that there are always other constraints for the design. Using a larger engine can result in higher thrust, but this increases the mass of the vehicle, making it less efficient. Similarly, using a higher chamber pressure leads to higher thrust, but makes the engine more complex.

Table 8.4: Nominal input and output parameters for FSSC-1 rocket engine sensitivity analysis (Daimler-Benz Aerospace, 1996c). Reference altitude for ambient pressure is 15 km.

Parameter	Value	Unit
	INPUT	
OF	6.6	-
A_e	1.570322	m^2
$A_t h$	0.104688	m^2
A_{ratio}	15	-
p_c	95	bar
p_a	12111	Pa
	OUTPUT	
T	1815.80	kN
$\dot{m}$	439.12	$kg\,s^{-1}$
C_T	1.71	-
I_{sp}	419.35	s
M_e	3.36	-
u_e	3869.52	$m\,s^{-1}$
p_e	80043.18	Pa
$x_{c,j}^{(1)}$	$[0.8755, 0.0170, 0.0271, 0.0718, 0.0061, 0.0023, 0.0002, 0.0000]$	$kg\,mol\,kg^{-1}$

(1) $j = [H_2O, O_2, H_2, OH, O, H, HO_2, H_2O_2]$

Table 8.5: Propulsion system sensitivities in percentages for propellant combination LO_2 + LH_2 for a range of 10% around the nominal input parameters. Reference vehicle is the FSSC-1 vehicle. Nominal conditions given in Table 8.4. The red cells indicate sensitivities larger than 10%.

	OF		p_c		A_e		A_{th}		p_a	
T	-3.01	1.87	-10.06	10.06	-9.46	9.36	-0.65	0.55	-0.11	0.11
$\dot{m}$	-4.50	2.39	-9.91	9.90	-10	10	-0.01	0	0	0
C_T	-0.61	0.23	-0.023	0.025	-0.94	1.01	-1.04	0.91	0	0
I_{sp}	-1.63	1.57	-0.17	0.14	-0.58	0.60	-0.64	0.55	-0.11	0.11
M_e	-1.87	2.45	-0.12	0.11	-1.82	2.03	-2.02	1.84	0	0
u_e	-2.10	1.69	-0.073	0.065	-0.94	1.01	-1.04	0.91	0	0
p_e	-6.98	7.05	-9.72	9.69	-12.90	13.33	-11.75	14.84	0	0
T_{cc}	-2.25	1.07	-0.37	0.33	0	0	0	0	0	0
x_{c,H_2O}	-3.46	2.57	-0.23	0.20	0	0	0	0	0	0
x_{c,O_2}	-61.51	108.33	-2.96	3.31	0	0	0	0	0	0
x_{c,H_2}	-29.01	41.10	-0.33	0.37	0	0	0	0	0	0
$x_{c,OH}$	-28.14	23.04	-1.30	1.42	0	0	0	0	0	0
$x_{c,O}$	-47.26	53.47	-3.43	3.86	0	0	0	0	0	0
$x_{c,H}$	-11.24	3.35	-2.41	2.70	0	0	0	0	0	0
x_{c,HO_2}	-56.24	81.60	-2.00	1.79	0	0	0	0	0	0
x_{c,H_2O_2}	-43.75	49.57	-5.20	4.90	0	0	0	0	0	0

Finally, increasing the mass flow is beneficial to the thrust, but again might require a larger propellant mass or engine, again creating different design drawbacks.

The same can be seen in Fig. 8.6. Most deviations are in the negative direction, with a few combinations that increase the parameters. In this case a larger magnitude of the variable does not necessarily mean 'better'. For example, a larger exit pressure is not always favourable as it means the engine is over or under expanded. This feature is visible in the graphs, since for exit pressure, most deviations are directed in the positive direction, making the exit pressure larger. The parameters, which are more directly related to the thrust, such as I_{sp}, u_e and C_T show the same behaviour as the thrust itself. The most interesting graph is the one for the combustion temperature, which does not look like a Gauss curve, as most of the others do. The rocket parameter results are very similar to the FSSC-12 case as well. Both cases use the same propellant combination, but have quite different dimensions and nominal outputs. Since the results are so similar, one could conclude that the model provides stable results for different conditions. Different propellants are analysed in the next section.

8.1.3. VERIFICATION TEST CASES

The three test cases presented in Table 7.1 are analysed in this section. These cases were chosen because they represent verified use cases, where the nominal in- and outputs are known. Furthermore, they give the opportunity of comparing different propellant combinations: LO_2 combined with LH_2, CH_4 or RP-1, which are the most commonly used bi-propellants. The effect of the propellant choice can more easily be identified due to the similar input parameters of these cases. Since the example cases do not mention the actual size of the nozzles or their mass flow, these parameters will not be investigated here. The focus is on the effect of the propellant selection. Since the mass flow and exit area are not included, the resulting thrust cannot be computed. Only the intermediate

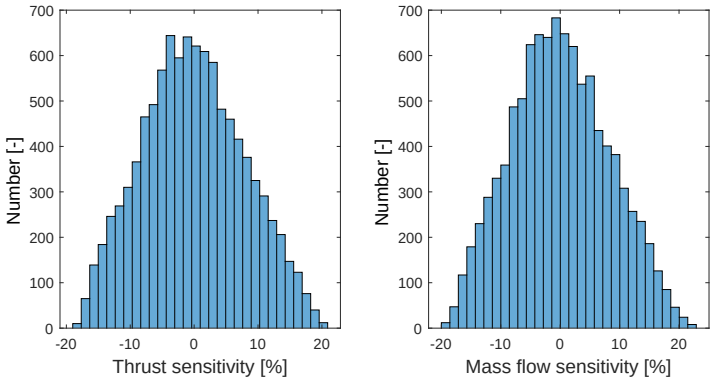


Figure 8.5: Thrust sensitivity resulting from a Monte Carlo simulation with 10000 runs, for an input variation of 10% for the FSSC-1 rocket engine.

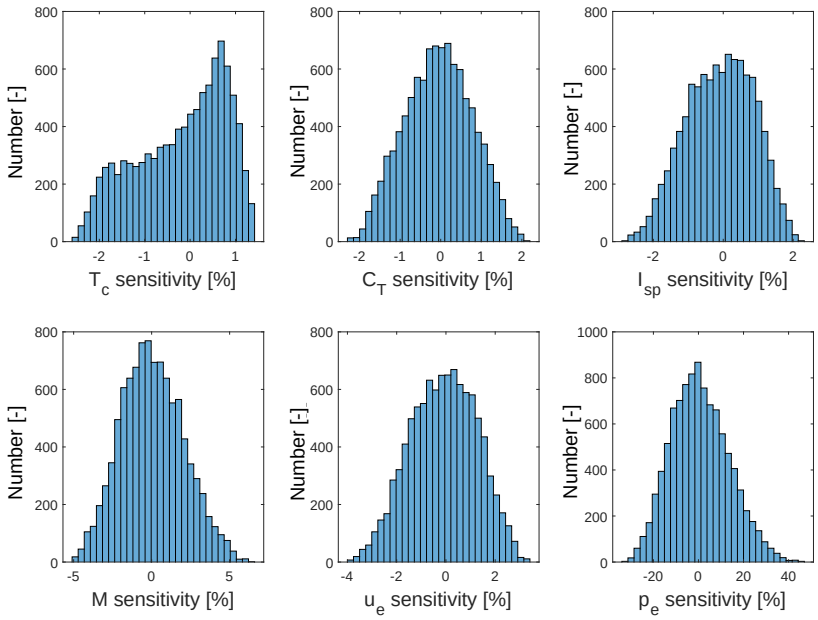


Figure 8.6: Rocket parameters sensitivity resulting from a Monte Carlo simulation with 10000 runs, for an input variation of 10% for the FSSC-1 rocket engine.

rocket parameters are investigated. The nominal in- and outputs are stated in Section 7.3.1. The sensitivity results for the three cases are given in Tables 8.6 to 8.8. Note that CH_4 and RP-1 have some additional combustion products.

Table 8.6: Propulsion system sensitivities in percentages for propellant combination LO_2 + LH_2 for a range of 10% around the nominal input parameters. Nominal conditions given in Tables 7.2 and 7.3. The red cells indicate sensitivities larger than 10%.

	OF		p_c		A_{ratio}		p_a	
C_T	-1.04	1.03	-0.039	0.044	-0.51	0.45	0	0
I_{sp}	-0.28	0.050	-0.52	0.43	-0.12	0.07	-0.46	0.46
M_e	-2.84	3.12	-0.071	0.063	-1.87	1.72	0	0
u_e	-0.60	0.35	-0.019	0.017	-0.51	0.45	0	0
p_e	-8.60	8.88	-9.82	9.81	-11.94	15.08	0	0
T_{cc}	-4.00	2.89	-0.27	0.24	0	0	0	0
x_{c,H_2O}	-1.32	0.38	-0.14	0.13	0	0	0	0
x_{c,O_2}	-69.84	163.25	-4.88	5.59	0	0	0	0
x_{c,H_2}	-24.64	31.47	-0.034	0.038	0	0	0	0
$x_{c,OH}$	-40.01	45.49	-2.21	2.45	0	0	0	0
$x_{c,O}$	-60.79	102.04	-4.99	5.73	0	0	0	0
$x_{c,H}$	-14.90	5.34	-2.82	3.17	0	0	0	0
x_{c,HO_2}	-67.26	138.80	-0.13	0.045	0	0	0	0
x_{c,H_2O_2}	-55.07	85.21	-3.75	3.46	0	0	0	0

First of all, it is clear that the overall sensitivities are relatively similar among these three cases, and even when comparing to the results of FSSC-1 and FSSC-12. This again proves the validity of the model, and that it functions well in many conditions. This also indicates that the sensitivity of the outputs does not have a large dependency on the chosen inputs, which makes it possible to generalise these results to a more broad context.

The OF ratio is the most influential parameter, and p_e is the most dependent (not taking in account the mole fractions). The mole fractions show similar behaviour, where most fractions have a high sensitivity, except for the main constituent of the exhaust gas: H_2O. However, it should be noted that for the CH_4 and RP-1 cases, the exhaust gas is more distributed over all constituents, as can be seen in Table 7.3. Also, for the first example case, H_2 makes up about a third of the exhaust gases. Looking at the main constituents except for H_2O (H_2, CO, CO_2), there sensitivities are quite large, up to 30%. This is very high, considering the inputs only vary 10%. This behaviour was seen in the previous examples as well.

The MC results are shown in Figs. 8.7 to 8.9. They do not vary a lot with respect to earlier results, and show a similar trend towards the negative. The sensitivities are definitely within bounds, and do not show clear outliers. p_e and M have the highest sensitivities, and the overall shape of the curves is comparable for all cases. This is good as it proves that using different propellants does not influence the sensitivities. A change of propellant does have large consequences for other parts of the vehicle design. For example, the LH_2/LO_2 combination has the highest I_{sp} or u_e, which is beneficial. A drawback of this propellant combination is that LH_2 must be stored at extremely low temperatures, which increases the complexity of the tanks.

Methane (CH_4) can be stored at room temperature, which is a main advantage with respect to LH_2. The density or energy density of the propellants play a role as well, since lower mass is favourable for any space design. RP-1 has the largest impulse density of the three combinations, and thus requires less stored propellant for the same energy. To

Table 8.7: Propulsion system sensitivities in percentages for propellant combination LO_2 + CH_4 for a range of 10% around the nominal input parameters. Nominal conditions given in Tables 7.2 and 7.3. The red cells indicate sensitivities larger than 10%.

	OF		p_c		A_{ratio}		p_a	
C_T	-1.72	1.38	-0.053	0.059	-0.57	0.50	0	0
I_{sp}	-1.56	0.61	-0.52	0.43	-0.050	0.0052	-0.45	0.45
M_e	-4.77	4.25	-0.12	0.11	-1.70	1.55	0	0
u_e	-1.13	0.17	-0.040	0.035	-0.57	0.50	0	0
p_e	-10.08	13.10	-9.73	9.71	-11.43	14.38	0	0
T_c	-3.13	1.19	-0.36	0.33	0	0	0	0
$x_{c,CO}$	-18.66	22.89	-0.20	0.22	0	0	0	0
x_{c,CO_2}	-16.47	12.46	-0.37	0.34	0	0	0	0
$x_{c,COOH}$	-20.06	3.95	-7.70	7.49	0	0	0	0
$x_{c,H}$	-10.92	1.77	-2.52	2.83	0	0	0	0
x_{c,H_2}	-31.09	53.06	-0.51	0.57	0	0	0	0
x_{c,H_2O}	-4.05	2.00	-0.25	0.22	0	0	0	0
x_{c,H_2O_2}	-54.35	54.44	-5.39	5.10	0	0	0	0
$x_{c,HCO}$	-27.62	29.60	-6.13	5.87	0	0	0	0
x_{c,HO_2}	-67.78	93.24	-2.29	2.06	0	0	0	0
$x_{c,O}$	-59.63	63.37	-3.27	3.68	0	0	0	0
x_{c,O_2}	-72.95	127.01	-2.61	2.90	0	0	0	0
$x_{c,OH}$	-37.91	26.71	-1.21	1.32	0	0	0	0

Table 8.8: Propulsion system sensitivities in percentages for propellant combination LO_2 + RP-1 for a range of 10% around the nominal input parameters. Nominal conditions given in Tables 7.2 and 7.3. The red cells indicate sensitivities larger than 10%.

	OF		p_c		A_{ratio}		p_a	
C_T	-1.79	1.17	-0.055	0.061	-0.59	0.51	0	0
I_{sp}	-1.65	0.58	-0.53	0.44	-0.039	0.0015	-0.44	0.44
M_e	-5.18	5.01	-0.15	0.14	-1.71	1.56	0	0
u_e	-1.15	0.13	-0.052	0.047	-0.59	0.51	0	0
p_e	-11.55	14.28	-9.66	9.63	-11.42	14.38	0	0
T_c	-2.33	0.82	-0.41	0.37	0	0	0	0
$x_{c,CO}$	-15.99	19.70	-0.20	0.22	0	0	0	0
x_{c,CO_2}	-16.66	12.23	-0.44	0.40	0	0	0	0
$x_{c,COOH}$	-15.66	2.57	-7.95	7.76	0	0	0	0
$x_{c,H}$	-11.63	4.49	-2.32	2.60	0	0	0	0
x_{c,H_2}	-28.38	48.33	-0.51	0.56	0	0	0	0
x_{c,H_2O}	-3.90	1.90	-0.33	0.30	0	0	0	0
x_{c,H_2O_2}	-47.22	42.03	-5.86	5.58	0	0	0	0
$x_{c,HCO}$	-25.26	29.74	-6.25	5.60	0	0	0	0
x_{c,HO_2}	-60.50	71.50	-2.90	2.65	0	0	0	0
$x_{c,O}$	-51.52	48.16	-2.79	3.13	0	0	0	0
x_{c,O_2}	-66.57	98.55	-2.07	2.29	0	0	0	0
$x_{c,OH}$	-31.33	20.27	-0.91	0.98	0	0	0	0

choose the appropriate propellants, a trade-off must be made, also including safety of hypothetical passengers, the environment, etc. The focus of this study is the LO_2/LH_2 combination, as these are used in all reference vehicles. However, it is interesting to investigate the implications of another bi-propellant.

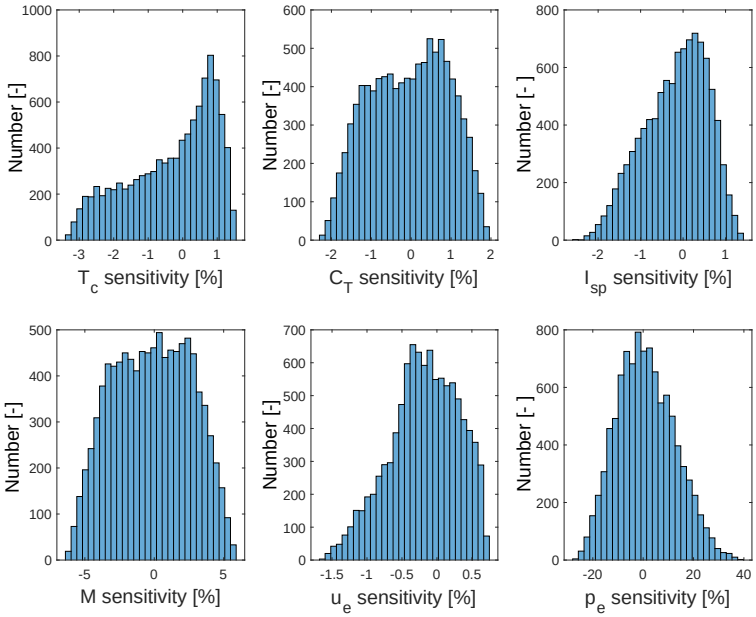


Figure 8.8: Rocket parameters sensitivity resulting from a Monte Carlo simulation with 10000 runs, for an input variation of 10% for the second verification case from Table 7.1.

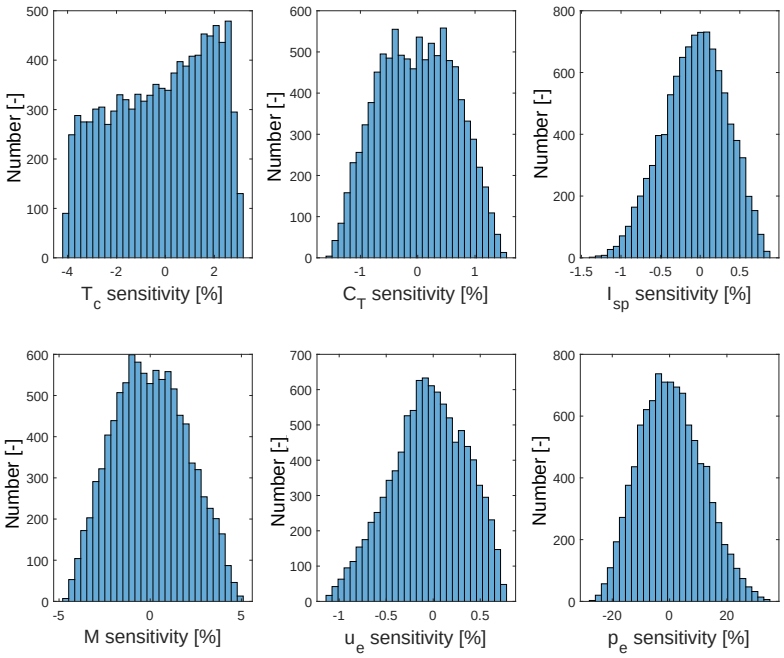


Figure 8.7: Rocket parameters sensitivity resulting from a Monte Carlo simulation with 10000 runs, for an input variation of 10% for the first verification case from Table 7.1.

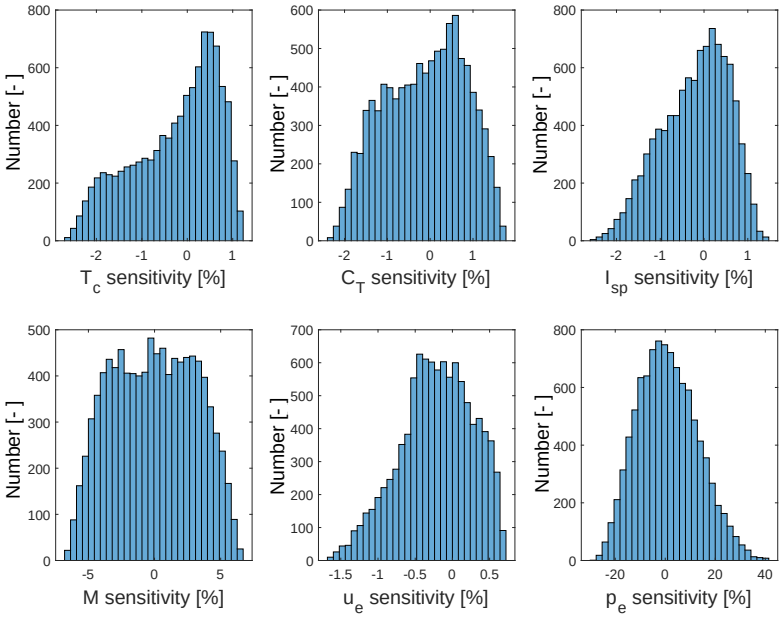


Figure 8.9: Rocket parameters sensitivity resulting from a Monte Carlo simulation with 10000 runs, for an input variation of 10% for the third verification case from Table 7.1.

To conclude: the thrust parameter sensitivities are generally around ±20% for input variations of ±10%, which is quite high. The important rocket parameters, such as the specific impulse and exhaust velocity, have small deviations. The thrust and mass flow have a linear relationship, which is to be expected from the thrust equation. This means any parameter that impacts the mass flow, will have a similar influence on the thrust force. Because of the fact that all input parameters, except for the ambient pressure, are inherent parameters of the rocket engine, and the resulting thrust is not affected by anything else, it is possible to study the stability and control of the vehicle with the thrust as input, instead of including the full propulsion model in the simulations. The main purpose of the further investigations will be to investigate the effect of the thrust on other parts of the system. Due to the analysis of this section, it is possible to effectively exclude the propulsion model from the full simulation model, and make the thrust or thrust range an input. This will decrease the run time of the simulations greatly, without reducing its accuracy a lot, as the thrust sensitivity is known at this point.

8.2. INERTIA MODEL SENSITIVITIES

The inertia model described in Sections 3.3 and 5.6 are analysed here to investigate the sensitivities of the vehicle inertia properties with respect to changing inputs. The main goal is to identify how sensitive the MMOI and COM values are with respect to the inclination angle of the propellants and the used propellant mass.

These are the two most important input parameters for this model. The angle has a large influence on the mass distribution along the vehicle, since it moves the propellant. The amount of mass used determines how much propellant is left in the tanks, and therefore has a large impact on the MMOI and COM. Furthermore, the more mass is used, the more the propellant can move in the tanks. For this purpose, a combined grid search and Monte Carlo analysis is used.

For several nominal inputs of the angle and used mass, a MC simulation with 1000 runs is carried out with random variations on the nominal inputs. This method is used, since it is important to determine the sensitivities at different set conditions, leading to the need for the grid search, after which the MC simulation is used to increase the efficiency of the program.

The used input parameter ranges for the grid search are given as follows:

$$M_{used} = [0.1, 0.3, 0.5, 0.7, 0.9] \times M_f \text{ (with } M_f \text{ the total propellant mass)}$$
$$\alpha = [-10, 0, 10, 20, 30, 40]°$$

The angle range from $-10°$ to $40°$ is chosen because this angle is almost always positive, and its range in the positive field will be much larger in reality.

For the MC simulations, the ranges were selected based on how the input parameters are expected to change during one simulation. Since the purpose of the simulations in the following sections is to analysis the short term reaction of the vehicle to some sort of step input or disturbance, most simulations will be quite short, not longer than 30 s. The step inputs in the commanded angles will be small as well, as will be explained in the coming sections. Therefore, both ranges can be kept relatively close to the nominal values. The mass range depends on the mass flow, and ranges between $\pm 30\dot{m}$, which translates to either 30 seconds of expelled propellant more or less than the nominal value. The angle range is taken as 5° to have some margin. The results are shown in Figs. 8.10 and 8.11. Note that symmetry along the y-axis is assumed, as well as no bank angle.

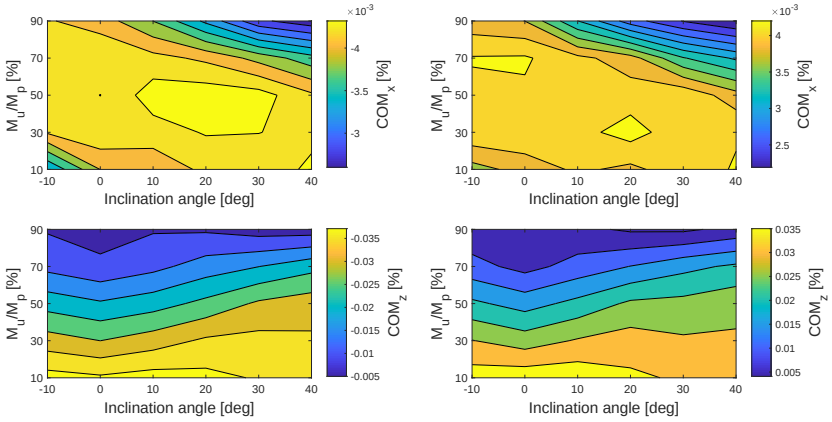


Figure 8.10: Centre of mass sensitivity for different inclination angles and at different mass stages. Each data point represents the minimum (left) or maximum (right) value for the sensitivity of all MC simulations. The sensitivities are expressed in percentages with respect to the nominal values.

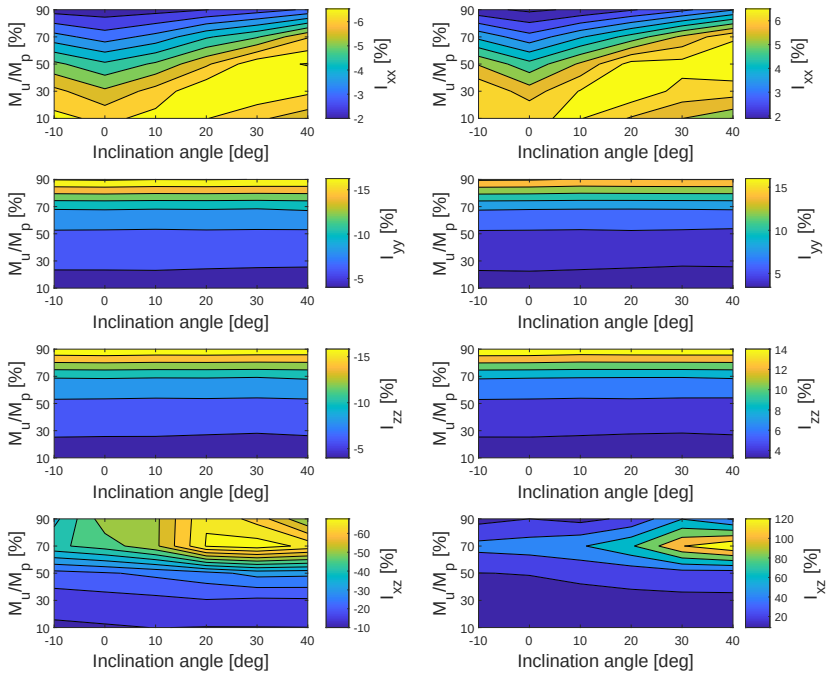


Figure 8.11: Moments of inertia sensitivity for different inclination angles and at different mass stages. Each data point represents the minimum (left) or maximum (right) value for the sensitivity of all MC simulations. The sensitivities are expressed in percentages with respect to the nominal values.

The COM sensitivity is very low. The deviations are never higher than even 0.1%. For COM_x, this is because of two factors:

- When most of the propellant is still present in the vehicle, it cannot move a lot due to the limits of the tanks, which leads to relatively constant COM locations.
- When most of the propellant is gone, its small mass, even though it moves by a large distance, does not have a large impact on the complete vehicle.

Since both of these limiting cases logically do not impact the location of the COM, the maximum deviations of COM_x lie in the middle, which is also visible in the figure. For the z-axis, the situation is quite different. Still the deviations are really small, but the shape of the graphs is not at all similar to the ones for COM_x. In this direction, the deviations are mainly influenced by the amount of mass left, and not by the angle. This is due to the fact that however the vehicle is located, the propellant will go down when expelled, reducing COM_z. Note that the reason for these small sensitivities is also the small deviations from the nominal conditions that are used here. The purpose is to obtain a

measure of the sensitivity over a short simulation period. This to determine whether these values can be kept constant and the model can be taken out of the simulation loop to make the tool more efficient. These graphs should not be read as this is the total sensitivity of the COM for ranges in inclination angle from −10 to 40, but rather how sensitive the system is to small variations on the nominal conditions around these points.

The inertia sensitivity is very high. The inertia values increase with the square of the distance of the mass to the COM. The motion and mass of the propellant therefore have a large impact on the inertia. A small change in the angle can move the propellant several meters further when the tanks are almost empty. Note that the sensitivity of the diagonal inertia products remains bounded with a maximum of ±15%. However, the I_{xz} deviations are incredibly high.

Notable is the apparent independence of the I_{yy} and I_{zz} sensitivity on the inclination angle. For a constant nominal mass, these values do not show any sensitivity change with respect to this angle. These are very sensitive at low propellant masses, as the motion of the propellant has a large impact. However, at even lower propellant masses, the sensitivities will suddenly decrease again. This because at some point, the propellant just cannot influence the vehicle anymore.

The same simulations were done for other input parameter ranges. The results were very consistent, and showed large sensitivity of the inertia for even smaller ranges. These results are not included here, since they do not add information.

9

Simulation Results & Discussion

In the previous chapter, the individual model sensitivities were determined. Now, all the individual models and components can be integrated in the simulator. Section 9.1 describes the performance metrics that are used for the analysis. The nominal system is analysed in Section 9.2. It is important to know the nominal response before off-design conditions are examined, as a benchmark is required. Afterwards, the propulsion effects on the controllability of the spaceplane are investigated in Section 9.3. Finally, measurement noise linked to the INDI controller is analysed in Section 9.4

In the following analysis, step commands of 1 degree are used to assess the performance of the vehicle. One degree commands are small, and perhaps not the best measure to examine the control behaviour. Note that tests with larger angles were done as well, to check if the controller could handle these. During ascent, the trajectory does not change very much, as indicated by Table 2.3. Although the angle of attack does change from 20° to 7.58°, this does not happen in one manoeuvre. Therefore, one degree commands are realistic in terms of the reference mission. During ascent, the main purpose of the control system is to provide stability to the vehicle with respect to disturbances. The same holds for sideslip and bank angle, as these are zero throughout the ascent. For deficiencies such as engine-out capabilities, it does not strictly matter if the vehicle can still perform large angle commands. It should however remain stable when this happens.

9.1. Performance Metrics

The results of the simulations can be presented in multiple ways. It is important to determine a set of performance metrics, which can be used to analyse the results in a straightforward way. Ogata (2010) describes several parameters that are often used to characterise the response of a system to an input:

- *Delay time*: The time required for the response to reach half of the final value the first time. For a 1° step input, the delay time would be the time at which the response crosses the 0.5° mark for the first time. It does not really give an indication of the final value of the response, or even whether the response satisfies the command.

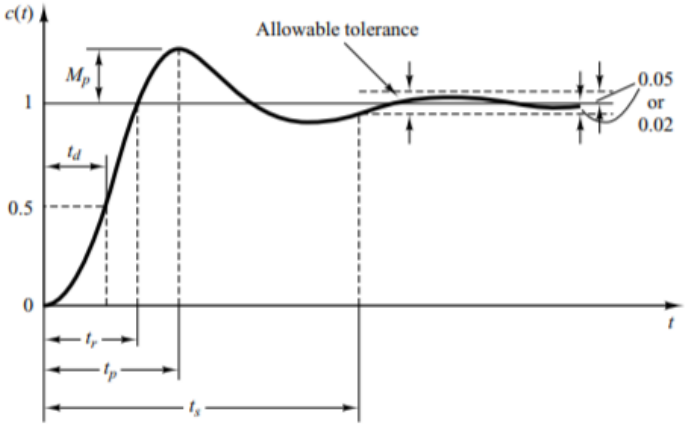


Figure 9.1: Representation of the response performance metrics (Ogata, 2010).

- *Rise time*: The time required for the response to rise from 10 % to 90 % of its final value.
- *Peak time*: The time required for the response to reach the first peak of the overshoot.
- *Maximum overshoot*: The maximum peak value of the response.
- *Settling time*: The time required for the response to reach and stay within 2 % of the final value. This is arguably the most important parameter, as it indicates whether the response is stable or not.

The five parameters are illustrated in Fig. 9.1

Next, Mooij (2016) and Viavattene (2018) describe several other metrics that can be used to analyse the response. For the control of the aerodynamic angles using moment fractions, these metrics are given as follows:

$$\sum_{\alpha_{err}} = \int_0^t \sqrt{(\alpha_c - \alpha_p)^2}\, dt, \quad \sum_{\beta_{err}} = \int_0^t \sqrt{(\beta_c - \beta_p)^2}\, dt, \quad \sum_{\sigma_{err}} = \int_0^t \sqrt{(\sigma_c - \sigma_p)^2}\, dt \tag{9.1a}$$

$$\sum_{\eta_x} = \int_0^t |\eta_x|\, dt, \quad \sum_{\eta_y} = \int_0^t |\eta_x|\, dt, \quad \sum_{\eta_z} = \int_0^t |\eta_x|\, dt \tag{9.1b}$$

where p and c stand for 'plant' and 'command', respectively. The equations above give a better representation of the complete response of the vehicle to a disturbance or command. By integrating the variables, not only does one get a sense of the magnitude of the response, it also helps to determine whether the response is stable, and provides an efficient way to compare multiple responses. In the figures, the $\sum$ sign is replaced by '*int*' (e.g. $\sum_{\alpha_{err}} = \text{int}\alpha_{err}$).

Apart form these metrics, the response of the vehicle to step commands is also compared with the requirement given in Fig. 2.2. Only when the response is not within this envelope, it will be mentioned in the report. The envelope specifies that the commanded angle cannot go down for the command. This criterion is not always satisfied as the vehicle motion often makes a small correction at the start of the simulation. This small disturbance is ignored with respect to this requirement.

9.2. NOMINAL SYSTEM ANALYSIS

The nominal model of the vehicle is defined as being the condition where all of the components function properly, without any errors or deviations from the norm. For example, the thrust force is constant, and directed exactly in the commanded direction. The inertia is computed perfectly, and all variables are known without any errors or noise. Furthermore, there are no disturbances, such as wind forces, or any other external moments. To assess the nominal model, the response of the system to a step input of 1° and 1° for the angle of attack and bank angle is determined. The five degrees input is included to check if the system can actually handle more than just one degree angles. The results are shown in Figures 9.2 to 9.4. The nominal trajectories are given in Table 2.3. The nominal configuration is given in Section 2.3.1.

Clearly, the system is capable of delivering a good response to the commands. After about 5 seconds, the commanded angles are reached. This conclusion is supported by the plots of the settling time, which are always close to or below 5 s. The size of the step does not have a large influence on the shape of the response, its rise time or settling time. The main difference is visible in the int$\eta_{y_{err}}$ graphs, where the integrated moment fraction for a higher step is larger as well. This is to be expected, since a larger step input requires larger torques. The intα_{err} graphs show a large difference as well. However, if both would be normalised (e.g. by dividing the one for the 5° step by five), approximately the same result would be obtained.

The settling and rise time plots show that the response is faster at higher Mach numbers. This could be due to several reasons. The most likely is the decrease of inertia at later flight stages due to the propellant expulsion. Lower inertia reduces the resistance against rotation, leading to faster responses. The second possibility is the increasing thrust at higher altitudes. As the atmospheric pressure reduces, the '$A_e\left(p_e - p_a\right)$'-term in the thrust equation increases, while the other part remains constant. A third reason could be the decrease of aerodynamic resistance at higher altitudes. However, since the aerodynamics scale with velocity squared, this is not always true, and depends on the specific case.

Similar conclusions can be drawn from the plots for the bank angle steps in Figs. 9.6 to 9.7. The reason for the similarity in the response at different trajectory points is due to the independence of the actuators, in this case the rocket engines, with respect to changing atmospheric conditions. Even though only the bank angle should change in an ideal case, the sideslip angle changes as well during this manoeuvre. This is due to the inherent connection between the lateral motion components. Therefore, a small integrated sideslip angle and corresponding η_z moment fraction are present as well.

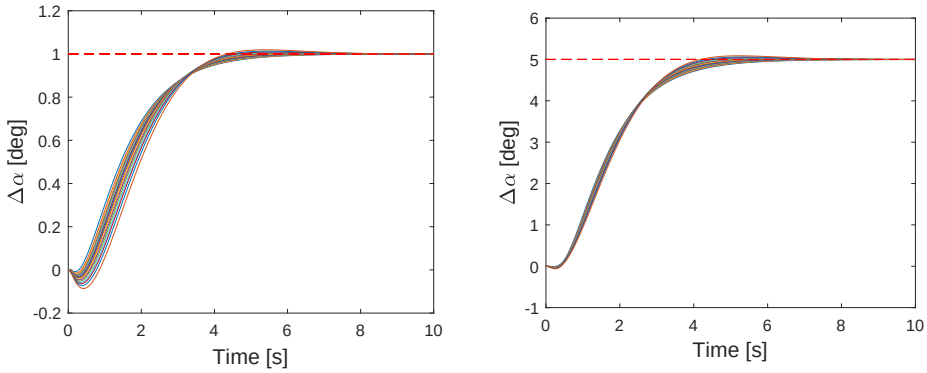


Figure 9.2: Nominal response to a 1 (left) and 5 (right) degree step input in angle of attack for all trajectory points from Table 2.3. Legend not included due to similarity of the lines.

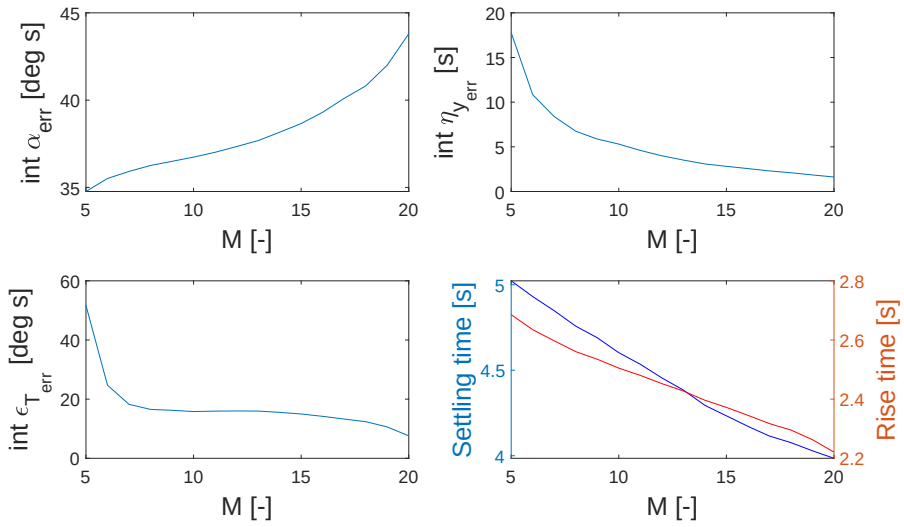


Figure 9.3: Nominal response performance indices to a 1-degree step input in angle of attack with respect to the Mach number.

Different step responses were tested as well, to get a better idea of the capabilities of the simulator. Negative step inputs, ramp inputs, and larger step angles were tested, all leading to similar results, which are not included here as they are redundant. It can be concluded that the simulator performs well in nominal conditions at any point in the vehicle's trajectory. These nominal values will be used to compare the non-ideal responses to in the coming sections.

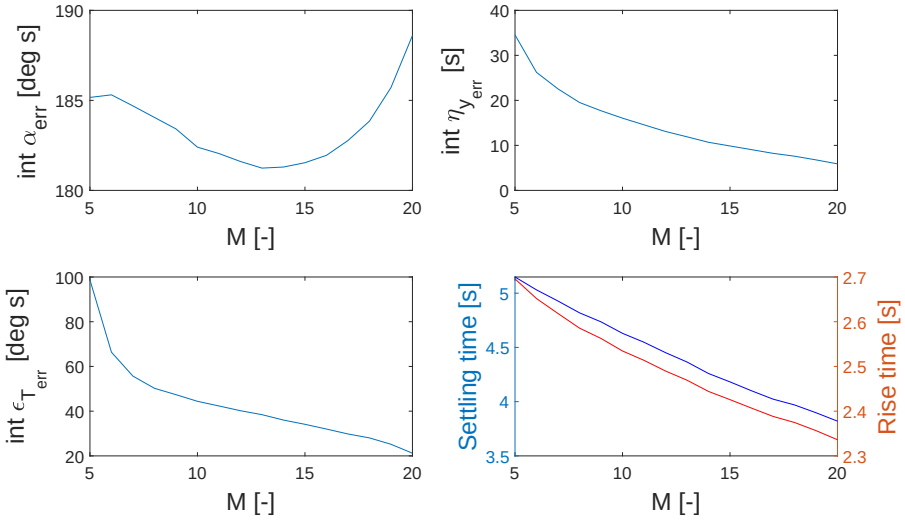


Figure 9.4: Nominal response performance indices to a 5-degree step input in angle of attack with respect to the Mach number.

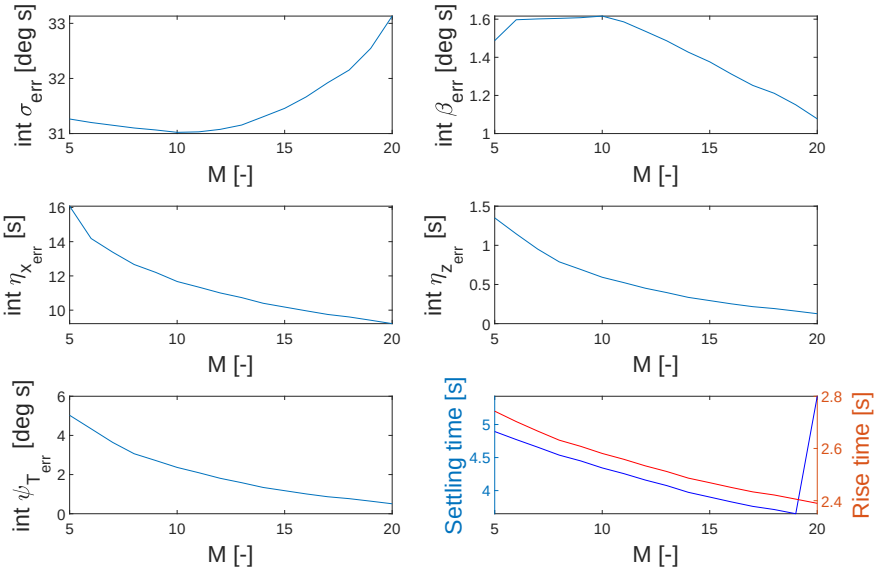


Figure 9.5: Nominal response performance indices to a 1-degree step input in bank angle with respect to the Mach number.

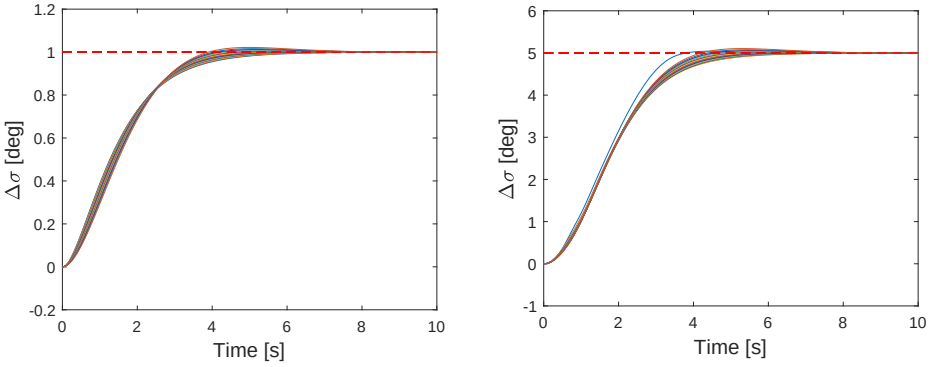


Figure 9.6: Nominal response to a 1 (left) and 5 (right) degree step input in bank angle for all trajectory points from Table 2.3. Legend not included due to similarity of the lines.

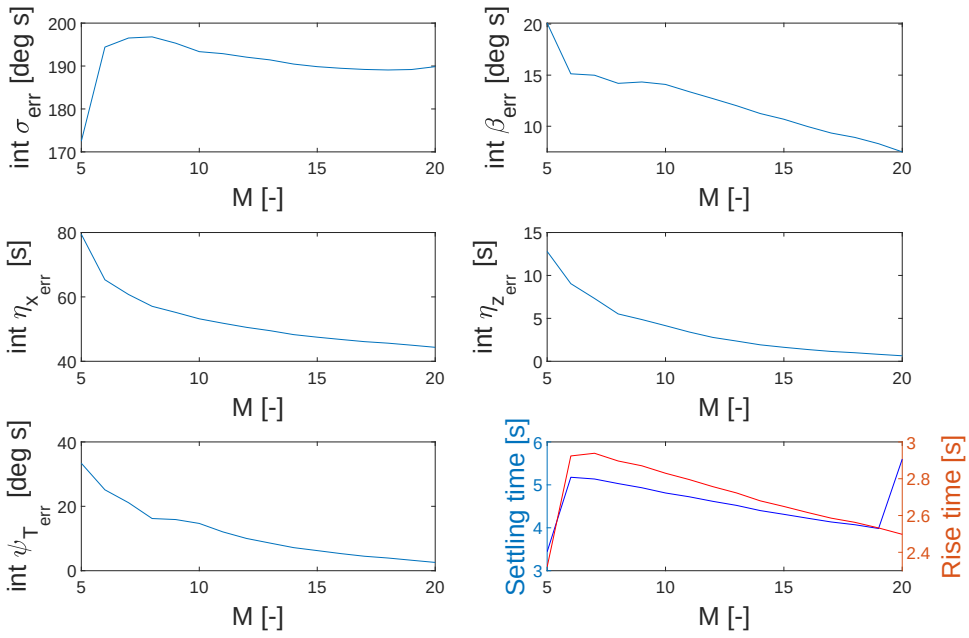


Figure 9.7: Nominal response performance indices to a 5-degree step input in bank angle with respect to the Mach number.

9.3. PROPULSION EFFECTS

This section includes all the results concerning the impact of the propulsion system on the response and controllability of the vehicle.

9.3.1. THRUST SENSITIVITY

As was established in Section 8.1, the thrust magnitude can vary a lot depending on the flight conditions. Therefore, it is necessary to simulate how the vehicle responds when the thrust magnitude is different. First of all, a Monte Carlo simulation is carried out, which computes the performance indices for 1000 runs. Each run has a constant thrust magnitude, ranging between ±20% of the nominal thrust value. These simulations were done for several trajectory points. However, since the results are similar, only the results of one trajectory point are reported here. Histograms of the performance parameters at $M = 15$ for a step input in angle of attack and bank angle are shown in Figs. 9.8 and 9.9, respectively.

First of all, one can notice that all simulations give similar results. The performance parameters do not change much depending on the trust magnitude. This is due to the fact that the TVC is extremely efficient. A small angle change of the thrust vector can create very high moments. This means the amount of thrust required for a small angle change is limited. Since the engines are located quite far from the COM, certainly in x-direction, the TVC provides a good way of controlling the motion of the vehicle. Moreover, it can be concluded that the created thrust is definitely large enough for TVC purposes. Despite a decrease of 20%, it can still control the vehicle adequately.

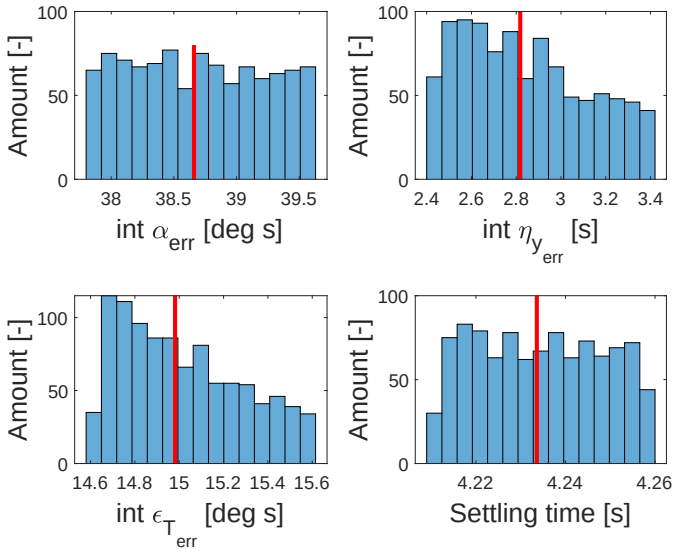


Figure 9.8: MC simulation with 1000 runs of a 1-degree angle of attack step input at $M = 15$ with a thrust magnitude variation of ±20% of the nominal value. The red lines indicates the nominal values.

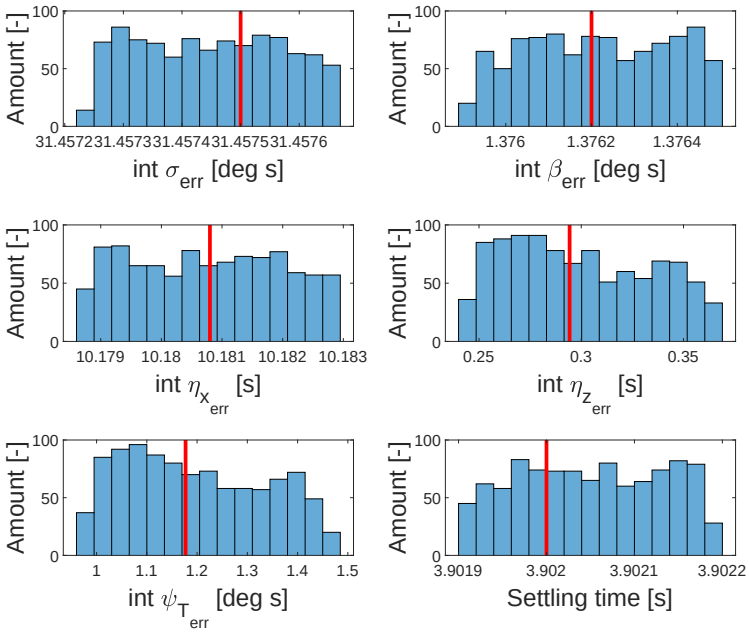


Figure 9.9: MC simulation with 1000 runs of a 1-degree bank angle step input at $M = 15$ with a thrust magnitude variation of ±20% of the nominal value. The red lines indicates the nominal values.

The magnitude of int$\epsilon_{T_{err}}$ is large for all simulations. This could be due to two different reasons; either a large control effort is required to deliver the 1-degree step in angle of attack, or the angle is not equal to zero even when no control moment is required. The second reason is true for this case. Since the rocket engines are used for trim, ϵ_T is non-zero most of the time. This is not the case for the lateral angle ψ_T, which leads to a far smaller value for int$\psi_{T_{err}}$. These same results were obtained for other trajectory points.

The previous simulations were done with a fixed thrust value for the complete simulation. An additional test was done with thrust fluctuations during one simulation. The thrust could fluctuate due to internal issues. For this, the thrust was simulated as a sine wave with the amplitude equal to the nominal thrust, and a variation of ±20%. The sine wave was given a period of 0.1/1/10/50 seconds. For each case, the step commands were satisfied without issue. The fluctuations were not even visible in the response, so they are not shown here. Only the thrust angle ϵ_T shows a minor fluctuation as shown in Fig. 9.10.

To conclude, the TVC can control the vehicle very well for many different flying conditions, and at changing thrust levels. Next, it is important to determine how the system reacts to an in-flight thrust magnitude change, which could be due to a different throttle setting, or the malfunction of one or more engines.

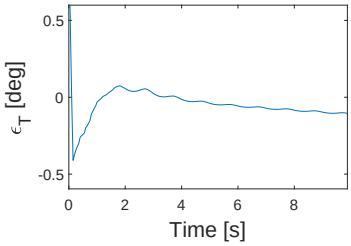


Figure 9.10: TVC ϵ_T command as response to a step command of 1 degree in angle of attack at M=15 including a thrust fluctuations with a period of 1 s.

9.3.2. IN-FLIGHT THRUST VARIATIONS

Originally, the vehicle operates with three rocket engines, each delivering the same thrust. Now, it is important to investigate how the vehicle reacts when an engine breaks down, or loses a part of its thrust. Another important factor to check is what happens when one of the engines gets stuck at a certain angle, and cannot move anymore due to some malfunction. This creates a constant disturbance moment, which could be unfavourable for the stability of the spaceplane.

First of all, the engine-out capabilities of the vehicle are examined. For this purpose, step commands of 1 degree in all three aerodynamic angles are simulated for four cases: all engines working, first engine off, second off and third off. Figures 9.11 and 9.12 show the angle response to the commands, and the TVC commands for the angle of attack step input.

The vehicle can perform the manoeuvre in all three engine-off cases. When the first engine is off, the commanded angle is reached more quickly than for the nominal case, because the engines are also used for trim stability. Evidently, the bottom engines are initially positioned in such way that they provide trim more efficiently. By adding the upper engine, they need more time to adjust for trim. For both bottom engines, the longitudinal motion is the same, and the lateral motion opposite. When one of the lower engines is cut off, the other creates a moment along the lateral axes, which is not compensated by the symmetrically opposed engine. Hence, β and σ experience a minor disturbance. The symmetry is also visible in the behaviour of ψ_T in Fig. 9.12. The fluctuations are quickly cancelled out by the TVC, however there seems to be a steady-state error. Although the TVC can provide stability to the vehicle in case off engine cut-offs, it cannot match the commanded angles exactly.

The results for the β-step simulations are given in Figs. 9.13 and 9.14. The plots show similar behaviour as for the α step command. The β command is matched and σ shows a steady-state error. The angle of attack plot shows a small disturbance at the start for all cases, which is due to trim. For this manoeuvre, ψ_T is close to being saturated right at the start for engine three off. Again, this means this manoeuvre will not be possible for engines with more strict motion limits. On the other hand, increasing the maximum allowable thrust angles, also increases the effectiveness of TVC. The drawback here is that these constraints usually depend on the physical motion limits of the engines.

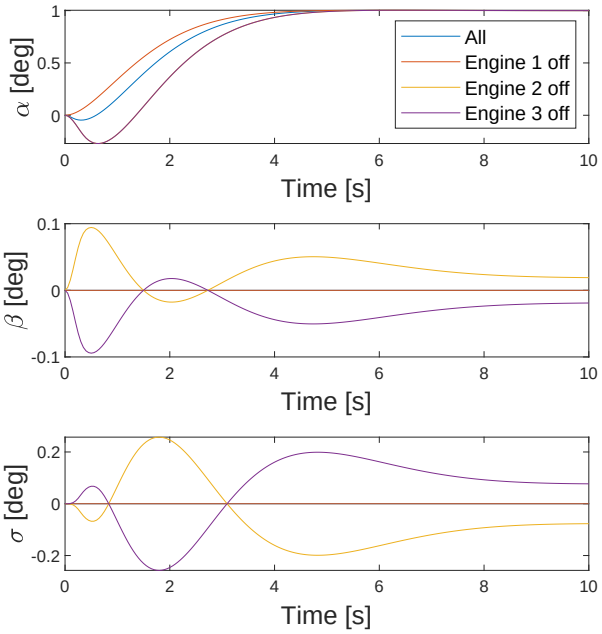


Figure 9.11: Response to a 1-degree step input in angle of attack for one engine out malfunctions.

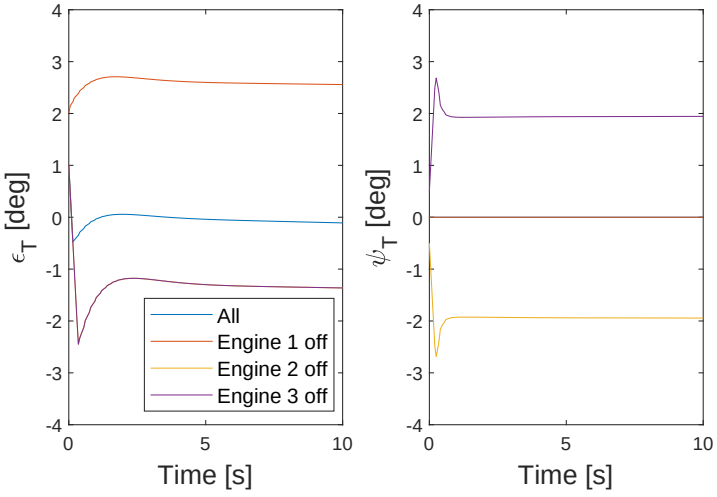


Figure 9.12: TVC commands for a 1-degree step input in angle of attack for one engine out malfunctions.

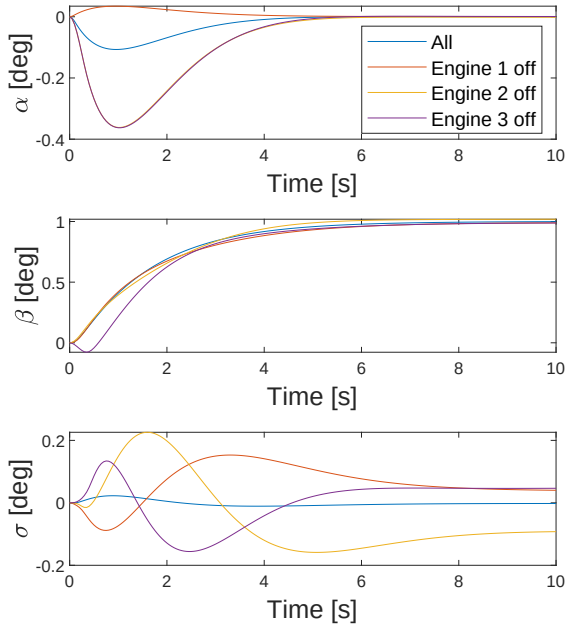


Figure 9.13: Response to a 1-degree step input in sideslip angle for one engine out malfunctions.

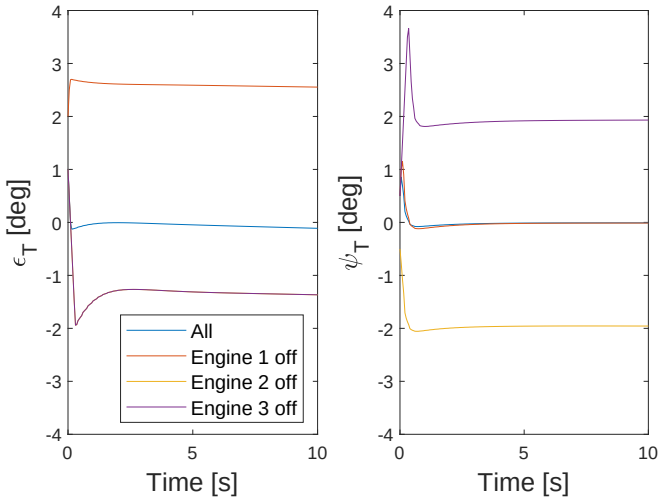


Figure 9.14: TVC commands for a 1-degree step input in sideslip angle for one engine out malfunctions.

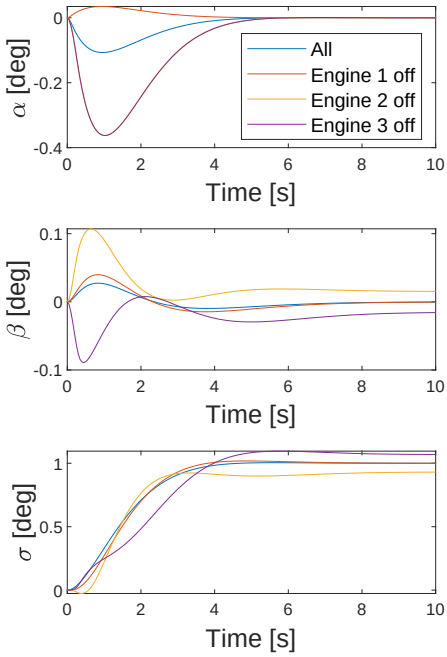


Figure 9.15: Response to a 1-degree step input in bank angle for one engine out malfunctions.

The bank angle command appears to be the most difficult to achieve for engine cut-off cases, as shown in Fig. 9.16. The command is in fact not realised, since a steady-state error remains. However, the stability maintained as there are no unwanted fluctuations after 10 s. The TVC angles show similar behaviour as for the other commands. The ϵ_T change is mainly for trim. This is indicated by the fact that the angle remains constant, different from zero, after a certain time. If not for trim, the angle would return to its initial value, or at least closer to it, after the manoeuvre is carried out. Although the controllability is still guaranteed in cases of engine cut-off, the same cannot be said for the translational motion and the success of the mission. With only two engines, the amount of thrust to propel the vehicle drops, and it might not be able to reach orbit anymore. This is out of the scope of this study, and is more a guidance issue. At least, this study proves that the mission could be aborted, and the vehicle could be saved by returning to ground in a controlled manner.

Another situation occurs when one of the engines gets stuck in a certain position and cannot change its thrust angles. The difference with the previous case is that this engine still creates thrust, and therefore a disturbing torque. In this case, the engine could be turned off if the disturbing moment is too large to overcome. Otherwise, if the disturbance is controllable by the other engines, the vehicle could keep its trajectory and achieve its mission. This depends on which engine gets stuck, and in what orientation.

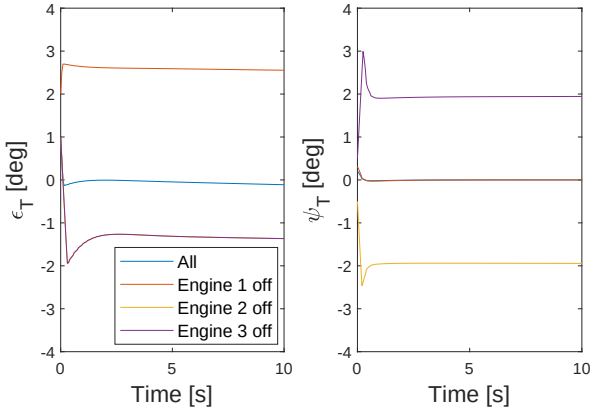


Figure 9.16: TVC commands for a 1-degree step input in bank angle for one engine out malfunctions.

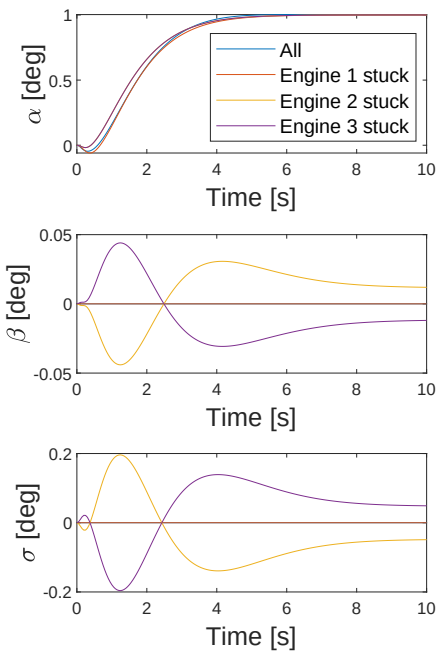
As a test case, all three engines are alternately fixed with $\epsilon_T = \psi_T = 0°$. Then, a step input in α and σ are simulated to examine whether the two remaining engines can control the vehicle. The results are given in Figs. 9.17a and 9.17b. For these cases, the step commands are satisfied within the 10 s time frame. There are some undesirable fluctuations in the response, but these are quickly damped. Again, cases involving engines 2&3 have steady-state errors.

Although these results look promising, different thrust angle combinations were investigated, and it was found that the stability is not guaranteed for all possible cases. The worst-case scenario with $\epsilon_T = \psi_T = 4°$ was not controllable for any of the engines. The main issue is the lateral control. When ψ_T is non-zero, the vehicle seems to have difficulty to control the large disturbance of the fixed engine. The longitudinal control shows a better response. The reason for the lateral problems is how the TVC is implemented in this study. As explained in 5.4.2, not the engines, but smaller thrusters are used for the roll control. For nominal operations, this method works perfectly, since the bank angle approximately remains zero throughout the ascent trajectory. However, when an engine gets stuck with $\psi_T \neq 0$, it immediately creates a relatively large roll moment, which cannot be controlled by the small thrusters.

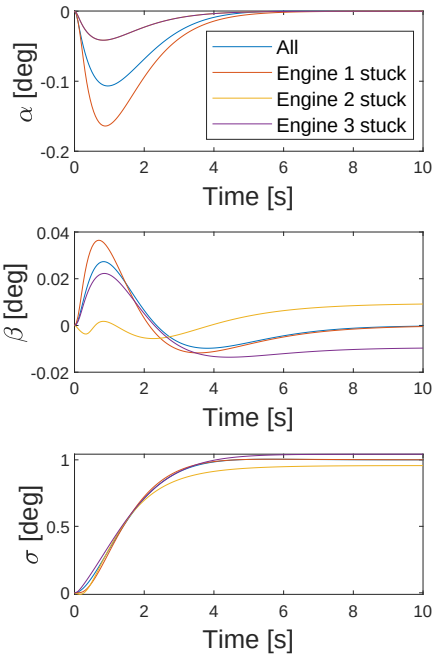
Possible solutions to this issue are:

- Use stronger roll thrusters.
- Implement a TVC strategy which uses the main engines for the roll control.
- Shut down the disturbing engine and abort mission in case the remaining thrust is not sufficient to reach the destination.

Option two is left for further research.



(a) Response to a 1-degree step input in angle of attack for engine stuck malfunctions.



(b) Response to a 1-degree step input in bank angle for engine stuck malfunctions.

9.3.3. PROPELLANT TANK EFFECTS

Not only the thrust and rocket engines can influence the controllability of the space-plane. The propellant tanks are an integral part of the propulsion system as well. The location of the tanks has a direct influence on the COM and inertia of the vehicle, and therefore indirectly on its controllability. Hence, it is important to study the effect of tank location on the stability of the vehicle. For this purpose, a MC simulation was carried out with 1000 runs. For each of the runs, the locations of the three tanks were varied within ± 1 m of their nominal position. This to determine the sensitivity of tank location on the performance metrics.

Fig. 9.18 gives the longitudinal performance metrics for a 1-degree step command in α. Two conclusions can be drawn from these results. Firstly, the settling time shows a symmetrical distribution, with a spread of about 1 s. This indicates that the command is realised in all of the tried simulations, and thus that this form of TVC can be used for different vehicle configurations. Secondly, the other performance metrics are not symmetrical at all. Most simulations are located on the left side of the graphs. However, there are still many outliers located towards the right. These indicate unfavourable behaviour of the controller for certain runs. For intα_{err} and int$\eta_{y_{err}}$, the outliers are reasonably close to the nominal value. For int$\epsilon_{T_{err}}$, this is not the case due to the need for trim.

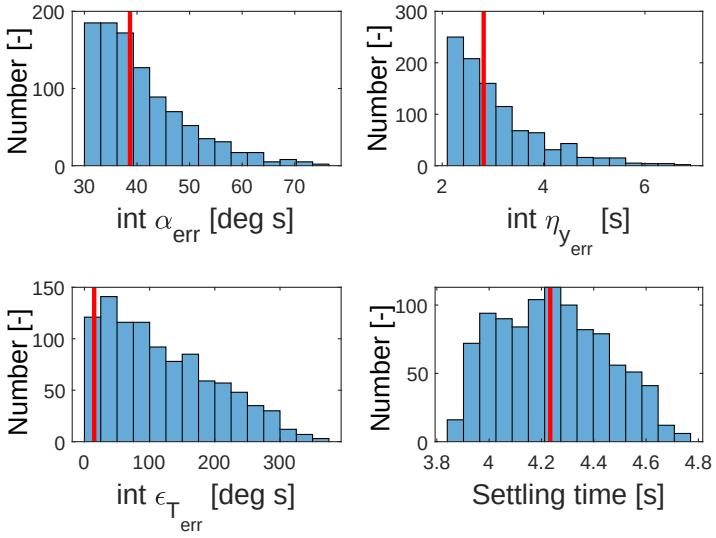


Figure 9.18: Monte Carlo simulation results for a 1-degree step command in angle of attack at $M = 15$ for varying tank locations.

For the nominal tank configuration, the required thrust deflection to realise trim stability is small. When the COM and inertia change, the trim requirements shift as well. Because of the near-optimal nominal configuration, this shift will be in the undesired direction in most cases. This is the reason for the non-symmetric distribution of the results.

The σ MC results are shown in Fig. 9.19. The results are much more symmetric than for longitudinal control. The nominal values are located in the centre of the histograms, and deviations remain small. This indicates the tank locations do not have a large effect on the roll control, meaning the nominal tank configuration is neither bad or especially good for yaw manoeuvres. The reason is the smaller inertia along the roll axis, and small distances between the x-axis and the tanks. It should be noted that changing the tank location in y- or z-direction is not equally realistic as in x-direction. This because propellant tanks are in essence always located symmetrically with respect to the y-axis, to avoid stability issues. Moreover, the tanks often take up the complete cross-section of the fuselage as they are so large, so a shift along the z-axis is mostly impossible. The reason these are included in the simulations is to broaden the scope of the research from one reference vehicle to a more general investigation of how these parameters can influence the control of a vehicle. Finally, the results for the β step command are given in Fig. 9.20. The results show some symmetry, but not as much as the previous case. This again indicates the nominal configuration is already somewhat optimised, which should of course be expected.

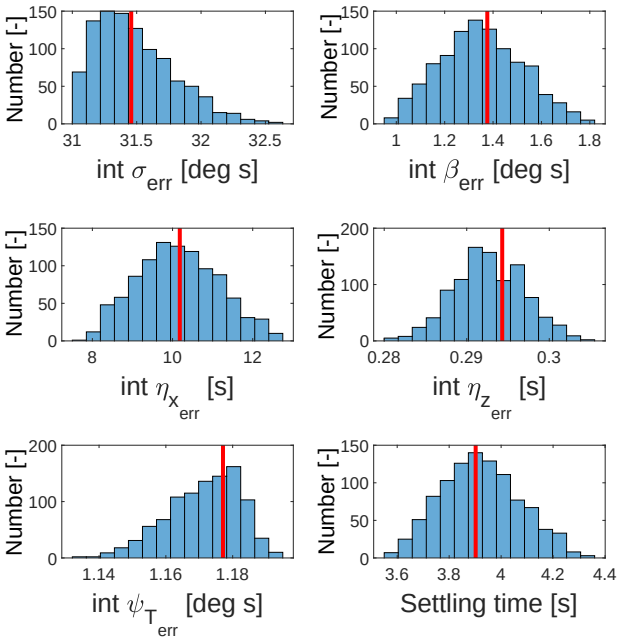


Figure 9.19: Monte Carlo simulation results for a 1-degree step command in bank angle at $M = 15$ for varying tank locations.

From these simulations, it can be concluded that the TVC control based on NDI manages to control the vehicle even when tank locations are shifted by as much as one meter. In all cases, the commands are satisfied within 10 s. So, although tank locations have an influence on the stability of the vehicle, a small shift will not have a detrimental impact on the controllability.

As explained in Section 5.6, the tanks can be emptied in different ways. The reference vehicle has three tanks: two with Oxygen, one with Hydrogen. The Oxygen tanks can be emptied at the same time (*[1,2],3*), or in a sequence (*1,2,3* or *2,1,3*). The sequence has a large influence on the MMOI during flight. This subject is briefly analysed to investigate its impact on the controllability of the vehicle. Only one of the trajectory points is analysed, namely $M = 15$. The results for the step responses are given in Fig. 9.21. The commands are satisfied for all cases. The α response shows undesired behaviour, as a large deflection in negative direction is visible for case 2 and 3. This is due to two reasons:

- The trim stability is more difficult to achieve for asymmetric emptying cases, since the MMOI and COM worse values.
- The control gains were determined with trial and error for the nominal case. This means the set of gains is not optimised for off-design situations, such as these.

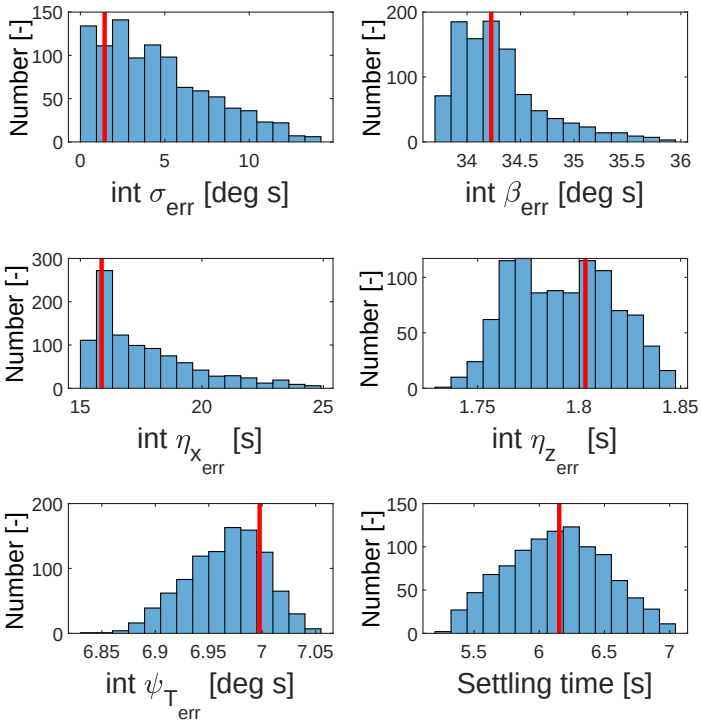


Figure 9.20: Monte Carlo simulation results for a 1-degree step command in sideslip angle at $M = 15$ for varying tank locations.

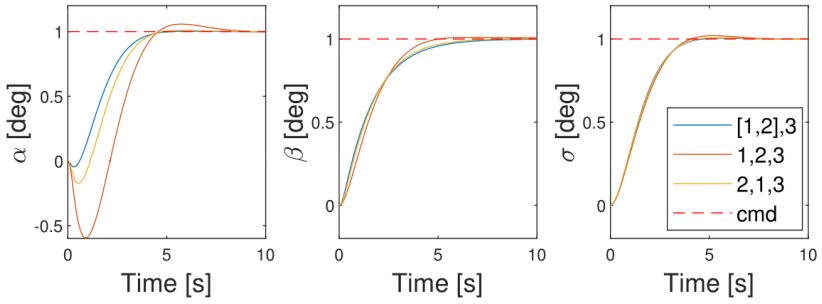


Figure 9.21: Response to a step command of 1° in angle of attack (left), sideslip angle (middle) and bank angle (right) at $M = 15$ along the ascent trajectory for three tank emptying cases.

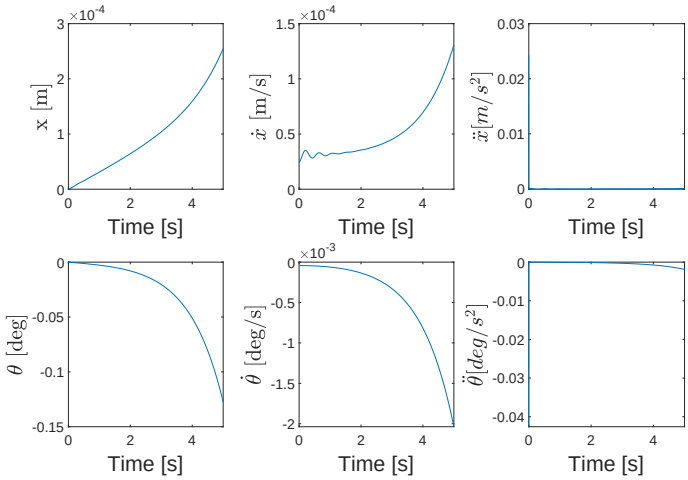


Figure 9.22: Vehicle response to an impulse force in x-direction of 1000 N.

SLOSHING

During the development of the sloshing models, several problems arose. First of all, the model equations in different references were all slightly different, without a good way to distinct the errors from the right equations. Furthermore, most references are not clear on their reference frames and on the limits of the models. The available time to develop the sloshing models was limited, so not much research could be done on this subject. Therefore, it must be said that even though parts of the model were verified, it is probable that the used models still contain small errors. The sloshing model was applied to the front LO_2 tank of FSSC-12. Two cases are analysed: one the response to an impulse force in x-direction, and the other the response to a fluctuating force of smaller magnitude. The results are shown in Figs. 9.22 to 9.25.

For both cases, the motion of the vehicle and slosh masses seem to be unstable after a short period of time. For the impulse input, some fluctuations are visible, which are damped out as expected. Afterwards, the motion appears to diverge very quickly. Similar results are obtained for the sinusoidal input, except that the sloshing motion is not damped out, since it is excited continuously.

The divergence of the results is unexpected and difficult to explain. The similar results were obtained for other test cases, with varying input magnitudes, directions, and other tank shapes. Similarly, the model diverges for inputs of M_y. There are a few reasons for the strange results. First of all, the model could contain errors that were not found during the verification process. This is plausible because of the errors found in reference documents. Secondly, the model could be correct, and the results would then indicate that this system is just not stable, and a control action is required. Sloshing is a complicated control problem, so this option can certainly be true. Finally, there is a possibility that the assumptions made are incorrect, and do not apply to this case. At this point, it cannot be concluded which of the reasons is correct. Therefore, the results

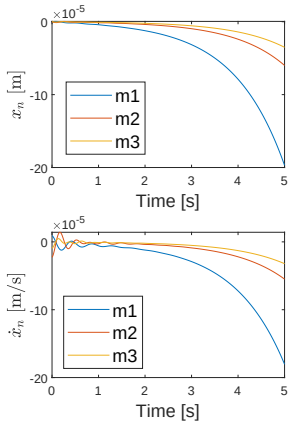


Figure 9.23: Sloshing response to an impulse force in x-direction of 1000 N for the first three sloshing modes.

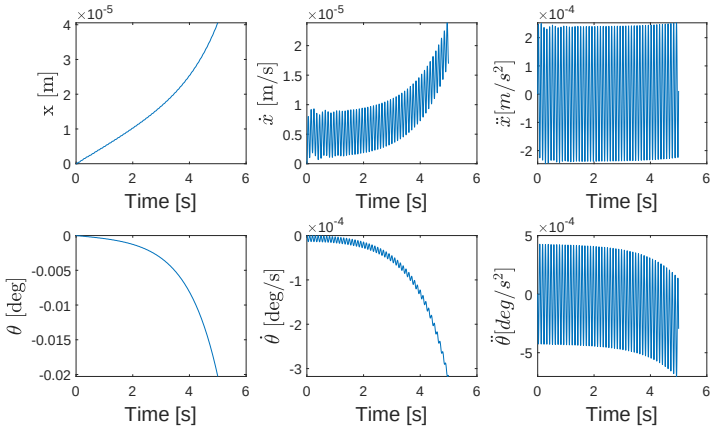


Figure 9.24: Vehicle response to a sinusoidal input force in x-direction with an amplitude of 10 N.

presented in this section should be treated with care. There is definitely much room for improvement on the sloshing model, and should be investigated more in the future.

In case the results presented in this chapter are correct, it means the sloshing is definitely an important factor that should be included in the control design of future missions. Furthermore, extra damping measures would be beneficial to limit the sloshing motion in the tank. Baffles are often used in propellant tanks to split the tank in sections. This effectively limits the sloshing motion, and improves the sloshing characteristics.

Since the used model does not lead to any sound conclusions, some additional simplifications were applied to the models to get an idea of the sloshing forces and moments that occur in certain conditions.

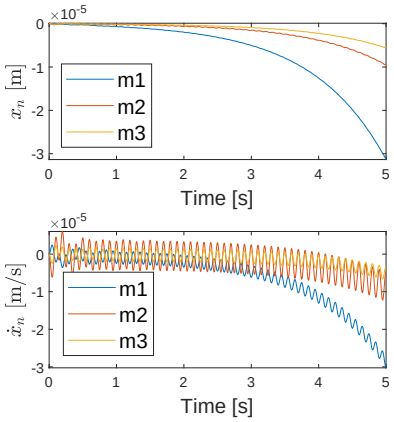


Figure 9.25: Sloshing response to a sinusoidal input force in x-direction with an amplitude of 10 N for the first three sloshing modes.

9.4. INDI MEASUREMENT NOISE

Normal NDI methods rely on dynamic models to compute the commands. The incremental form relies on sensor measurements of $\dot{\omega}$ to compute an increment to the control moment of the previous time step. Until this point, all measurements were assumed to be perfect, without any noise or biases. The inclusion of noise is investigated briefly for completeness. Only a step command in angle of attack is examined for three cases: noise level of $0.01\,^\circ s^{-1}$, $0.1\,^\circ s^{-1}$ and $1\,^\circ s^{-1}$. Modern gyroscopes have an accuracy of less than $0.01\,^\circ s^{-1}$, so these noise levels are used mainly as an example. Wit lower noise, the INDI controller can still satisfy the command accurately. However, at $1\,^\circ s^{-1}$, the response shows undesired fluctuations. The motion is still somewhat controlled, but only because all other conditions are perfect. The same fluctuations are visible for other manoeuvres, but not shown in this report.

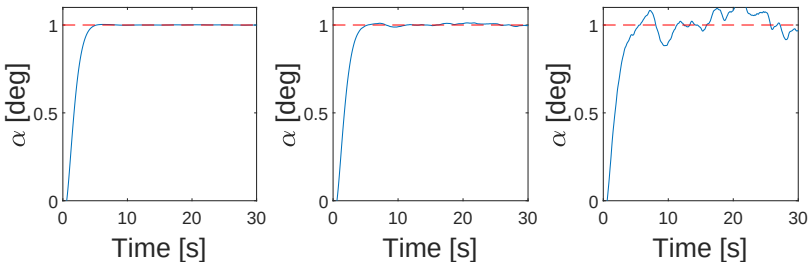


Figure 9.26: Response to a 1 degree step command in angle of attack at $M = 15$ along the reference trajectory, for different values of the noise factor in the $\dot{\omega}$ measurements. Noise levels equal to $0.01\,^\circ s^{-1}$ (left), $0.1\,^\circ s^{-1}$ (middle) and $1\,^\circ s^{-1}$ (right).

10

Conclusion & Recommendations

In Section 2.1, the main research question and several subquestions were given. The purpose of this chapter is to answer each of these questions, based on the information in this thesis report. A lot of research was done to analyse the motion of a spaceplane in hypersonic ascent trajectory. The sensitivities of the inertia and propulsion models were examined, the controllability of the vehicle at multiple design points was analysed, and the effect of the propulsion system deficiencies on the controllability of the spaceplane was investigated. Afterwards, some recommendations for future research are stated.

10.1. Conclusion

The main research question is as follows:

Which parameters and deficiencies in the propulsion system of a HORUS-type spaceplane effect its controllability at single points during the hypersonic ascent phase?

To answer this question on its own is challenging. Therefore, several subquestions were established to guide the research. Each of these is examined below.

The first two questions can be answered simultaneously, since they are very similar:

SUBQ-1: Which propulsion system parameters have the most effect on the resulting engine thrust?

SUBQ-2: What is the sensitivity of the thrust with respect to flight conditions and engine properties?

In Section 8.1, the thrust sensitivity with respect to the propulsion model input variables was examined. A thrust sensitivity of $\pm 20\%$ was found for input variable deviations of $\pm 10\%$. The combustion chamber pressure and nozzle exit area proved to be the most influential parameters on the thrust magnitude. The mole fractions are very sensitive to input variations as they are prone to computational errors because of their small values.

Since rocket thrust does not depend on many external parameters, it was decided that the full propulsion model could be kept outside the simulation loop.

Rocket performance is almost independent of the flight conditions. A rocket engine will perform similarly at sea-level or in vacuum. Only the ambient pressure influences the thrust magnitude, and even this variable only has a small influence on the thrust, as was shown in Chapter 8.

The same results were found for other engines and vehicles, which indicates that the thrust sensitivity does not have a high dependency on the engine properties, or even on the used propellants. Of course, the thrust magnitude depends heavily on these parameters, but its sensitivity is affected similarly for all cases.

SUBQ-3: How do tank location, size and inclination impact the mass moment of inertia of the spaceplane?

Section 8.2 describes the sensitivity of the tank inertia models with respect to the tank inclination angle and expelled mass. It was found that the tank inertia heavily depends on these variables, and is extremely volatile. The amount of propellant left in the tank has a logical influence on the inertia, since lower mass usually means smaller inertia. It was, however, interesting to investigate how the inclination angle impacted the inertia, since this is an aspect that is usually not included in most concept studies, and was not reported in any of the reference material. Looking at the results, it should be concluded that this is an aspect that can definitely not be overlooked during the design of a spaceplane.

The tank location and size have a direct connection with the inertia and COM values through the equations from Section 5.6. It is evident that larger tanks, including more propellants, will increase the moments of inertia. The same is true for increasing their distance to the centre of mass of the vehicle. On the one hand, increasing the MMOI is advantageous, since it increases the resistance against disturbances. On the other hand, larger MMOI also means the control efforts to perform a manoeuvre have to be increased. The mass distribution is important for the flying qualities and stability of the vehicle as well. However, this is a subject that has been studied in depth in the past.

SUBQ-4: Can TVC be used to perform manoeuvres at all design points along the trajectory?

In nominal conditions, TVC has the ability to control the vehicle at all design points as was shown in Section 9.2. TVC can perform step commands in angle of attack, bank angle and sideslip angle without any issues. The INDI controller, which was the main controller used for most of the simulations, shows robust performance in all tested conditions. It was found that the response is faster at higher altitudes, which could be due to the lower inertia or higher effective thrust in these regions.

The remaining two questions can also be answered at the same time:

SUBQ-5: Are the feedback control laws given in Chapter 6 sufficient to provide control in case of engine deficiencies?

SUBQ-6: Can the vehicle be controlled in off-design cases where part of the propulsion system fails?

The investigation of engine deficiencies was translated into several separate problems. First of all, there is the effect of thrust variations due to the thrust sensitivity described previously. Can the vehicle still perform the manoeuvres if the thrust changes by ±20%? The answer is yes. For all 1000 simulations with varying thrust magnitude, the response was satisfactory, and within bounds. The results are all so close it can be concluded that the TVC is extremely efficient and can function with less thrust than available. This is definitely beneficial, since it limits the required thrust angles. The more the engines have to move, the more chance failure occurs.

Next, engine failure cases were investigated. Firstly, the case when one of the engines completely ceases functioning was examined. For all three cut-off cases, the response is satisfactory, and the stability of the vehicle remains. Only small steady-state errors in the lateral angles are introduced when asymmetric engines are cut off. Secondly, the case where an engine gets stuck in one position, but keeps delivering thrust was examined. Again, the step commands were satisfied by the controller.

Finally, the impact of the tank locations on the response of the vehicle was investigated. This is more a design problem, rather than something that could occur during a mission. For future designs, knowing the impact of changing the mass distribution of the vehicle might prove important. It was identified that nominal configuration is relatively optimised for longitudinal control, since moving the tanks mostly had a negative impact on the performance metrics. For lateral motion, this was somewhat less. The influence of the tank locations is visible, but does not lead to uncontrollable motion.

10.2. RECOMMENDATIONS

This thesis contains many components and aspects, which means certain components have not been studied in as much detail as one would like. From this study, many conclusions have been drawn, but it is essential to elaborate on this study and investigate each of the components in more detail. Therefore, this section contains several recommendations for future work and research:

1. This thesis focuses on single design points in the ascent trajectory of the spaceplane. Although this method serves as a good basis for studying the controllability characteristics of the vehicle, it would be interesting to broaden the scope to the complete trajectory. For this, a dedicated guidance system should be added to the flight simulator, as well as the translational motion. Moreover, at this point no special manoeuvres are investigated, such as steady-level turns during cruise-flight. Most vehicles require at least some manoeuvres to reach their destination, and these often put the most stress on the control system.

2. Additionally to a guidance system, a navigation system should be added as well. This to make the simulations more realistic. No sensor is perfect. There are biases, noise and other factors influencing the measurements. The controller relies on the information it receives from these sensors. Hence, the controller can only work properly if the measurements are accurate.

3. The rocket engines are simulated as perfect actuators, meaning they can move in any direction as commanded by the control system, and as fast as required. Realistic actuators cannot move infinitely fast, and respond more slowly to the commands. This aspect should definitely be investigated in more detail, since it is visible in some of the results that the thrust angles move rapidly from one position to the other.

4. Sloshing was only treated briefly in this study, since time was limited. Investigating sloshing behaviour could be a thesis topic on its own, since it involves so many aspects. The mechanical model is appropriate for a quick analysis, but more complex CFD models should definitely be implemented in the future for more accurate studies.

5. The rocket engine was chosen in this study, since it is relatively simple compared to air-breathing engines. However, it would be interesting to do similar analyses for different engines, certainly because the spaceplanes of the future might rely on ram- or scramjet engines.

6. Only the INDI controller was used in this study. A comparative study of different controllers, as was done by Brinkman (2017) for the re-entry flight, is necessary to determine the most optimal controller. INDI is robust and easy-to-use, but might prove to be ineffective compared to other controllers.

BIBLIOGRAPHY

Abramson, N. H., "The dynamic behavior of liquids in moving containers." Southwest Research Institute, report NASA SP-106, 1966.

Ashford, D., *Spaceflight revolution*. Imperial College Press, 2002.

Basurto, E. R., Chen, P. Y., and Roberts, J. R., "Slosh design handbook - I," NASA, report NASA-CR-406, 1966.

Brinkman, D., "Knowledge-based control systems for re-entry vehicles," Faculty of Aerospace Engineering, Delft University of Technology, Msc. thesis, 2017.

Burcat, A. and Ruscic, B., "Third millenium ideal gas and condensed phase thermochemical database for combustion (with update from active thermochemical tables)," ; Argonne National Lab. (ANL), Argonne, IL (United States), report ANL-05/20, 2005.

Chu, P. Q., "Advanced flight control," Faculty of Aerospace Engineering, Delft University of Technology, Lecture slides, 2018.

Cohen, H., *Numerical Approximation Methods*. Springer, 2011.

Daimler-Benz Aerospace, "FSS-12 conceptual launcher design - evaluation and synthesis," Daimler-Benz Aerospace, Report FSS-SCT-RP-0007/2, 1996a.

——, "FSS-12 re-configuration and performance - evaluation and synthesis," Daimler-Benz Aerospace, Report FSS-SCT-RP-0045, 1996b.

——, "Structural and thermal design report slice C," Daimler-Benz Aerospace, Report FSS-SCT-RP-0053, 1998.

——, "FSS-1 conceptual launcher design - evaluation and synthesis," Daimler-Benz Aerospace, Report FSS-SCT-RP-0001, 1996c.

——, "Preliminary definition of the flight control subsystem (FCS) for the reference concept configurations no. 1 (FSS-1) and no. 9 (FSS-9)," Daimler-Benz Aerospace, Report FSS-SE-RP-0001, 1996d.

——, "Preliminary definition of the flight control subsystem (FCS) for the reference concept configurations no. 5 (FSS-5) and no. 13/14 (FSS-13/14)," Daimler-Benz Aerospace, Report FSS-SE-RP-0001, 1996E.

Dodge, F., *The New "dynamic Behavior of Liquids in Moving Containers"*. Southwest Research Inst., 2000.

Dujarric, C., "Possible future european launchers - a process of convergence," *ESA BULLETIN*, no. 97, pp. 11–19, 1999.

El-Sayed, A. F., *Fundamentals of Aircraft and Rocket Propulsion*. Springer, 2016.

Erdogan, K., "Spline interpolation techniques," *Journal of Technical Science and Technologies*, vol. 2, p. 47, 2014.

Hairer, E., Nørsett, S., and Wanner, G., *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer Berlin Heidelberg, 2008.

Hauser, J., *Numerical Methods for Nonlinear Engineering Models*. Springer Netherlands, 2009.

Koelle, D. E. and Kuczera, H., "Sänger II, an advanced launcher system for europe," *Acta Astronautica*, vol. 19, no. 1, pp. 63–72, 1989.

Kuczera, H., Hauck, H., Krammer, P., and Sacher, P., "The german hypersonics technology programme - status 1993 and perspectives," American Institute of Aeronautics and Astronautics, Conference paper, 1993.

Lindfield, G. and Penny, J., *Numerical Methods Using MATLAB*. Elsevier Science & Technology, 2012.

Marchi, C. and Araki, L., "Evaluation of chemical equilibrium and non-equilibrium properties for LOX/LH2 reaction schemes," *Journal of Aerospace Technology and Management*, vol. 7, pp. 31–42, 2015.

MBB , "Study on re-entry guidance and control - final report," MBB Space Communication and Propulsion Systems Division, conference paper ESA CR (P) 2652, 1988.

McBride, J. B. and Sanford, G., "Computer program for calculation of complex chemical equilibrium compositions and applications - I: Analysis," NASA, Report 0704-0188, 1994.

——, "Computer program for calculation of complex chemical equilibrium compositions and applications - II: Users manual and program description," NASA, Report 0704-0188, 1996.

Montenbruck, O. and Gill, E., *Satellite Orbits: Models, Methods, and Applications*. Springer Berlin Heidelberg, 2000.

Mooij, E., "Linear quadratic regulator design for an unpowered winged re-entry vehicle," Delft University of Technology, Report LR-806, 1997.

——, "Aerospace-plane flight dynamics: analysis of guidance and control concepts," Delft University of Technology, Phd. thesis, 1998.

——, "The motion of a vehicle in a planetary atmosphere," Delft University of Technology, Report LR-768, 1994.

——, "Re-entry systems," Faculty of Aerospace Engineering, Delft University of Technology, Lecture slides, 2016.

Mooij, E. and Ellenbroek, M. H. J., "Multi-functional guidance, navigation and control simulation environment - rapid prototyping of space simulations," Chapter 15 in: Rapid Prototyping Technology - Principles and Functional Requirements (Ed. Muhammad Enamul Hoque), InTech, September 2011.

Mooij, E. and Wijnands, Q. G. J., "Generic attitude and orbit control simulator development supporting the aocs software life cycle," From: 7th International Workshop on Simulation for European Space Programmes, Noordwijk, November 2002.

Mooij, E., "The Horus-2B reference vehicle," Delft University of Technology, Memorandum M-692, 05/1995 1995.

Mooij, E. and Gransden, D., "The effect of sloshing on the controllability of a conventional aeroelastic launch vehicle," report AIAA-2019-0116, 2019.

Mulder, J., van Staveren, W., van der Vaart, J., de Weerdt, E., de Visser, C., in 't Veld, A., and Mooij, E., "Flight dynamics," Faculty of Aerospace Engineering, Delft University of Technology, Lecture notes, 2013.

NASA, "U.S. standard atmosphere, 1976," NASA, Report NASA-TM-X-74335, 1976.

Nichkawde, C., Harish, P. M., and Ananthkrishnan, N., "Stability analysis of a multibody system model for coupled slosh–vehicle dynamics," *Journal of Sound and Vibration*, vol. 275, no. 3, pp. 1069–1083, 2004.

Ogata, K., *Modern Control Engineering*, 5th ed. Prentice Hall, 2010.

Ponomarenko, A., "RPA: Design tool for liquid rocket engine analysis," Rocket Propulsion Software+Engineering UG, Report, 2009.

Raouf, A. I., *Liquid Sloshing Dynamics: Theory and Applications*. Cambridge University Press, 2006.

Rijnsdorp, J., "Performance of a flush airdata sensor in a particle filter-based re-entry navigation system," Faculty of Aerospace Engineering, Delft University of Technology, Msc. thesis, 2017.

Sieberling, S., Chu, Q., and Mulder, J., "Robust flight control using incremental nonlinear dynamic inversion and angular acceleration prediction," *Journal of guidance, control and dynamics*, vol. 33, no. 6, 2010.

Viavattene, G., "Flying qualities and controllability of hypersonic spaceplanes," Faculty of Aerospace Engineering, Delft University of Technology, Msc. thesis, 2018.

Wang, X., Chu, Q., van Kampen, E., and Lu, P., "Stability analysis for incremental nonlinear dynamic inversion control," *Journal of guidance, control and dynamics*, vol. 42, no. 5, 2019.

Wu, S.-F., Engelen, C., Babuska, R., Chu, P. Q., and Mulder, J., "Intelligent flight controller design with fuzzy logic for an atmospheric re-entry vehicle," American Institute of Aeronautics and Astronautics, Tech. Rep., 2000.

"I think we've outgrown full-time education ... Time to test our talents in the real world, d'you reckon?"

Fred Weasley, *Harry Potter and the Order of the Phoenix*

A

HORUS-2B AERODYNAMIC MODEL

The aerodynamic coefficients are functions of the Mach number, angle of attack, altitude, sideslip angle and the control surface deflection angles. The aerodynamic force and moment coefficients are then defined as follows:

$$C_D = C_{D_0} + \Delta C_{D_{r,l}} + \Delta C_{D_{r,r}} + \Delta C_{D_{e,l}} + \Delta C_{D_{e,r}} + \Delta C_{D_{b,l}} - \Delta C_{D_h} \tag{A.1a}$$

$$C_S = \Delta C_{S_{r,l}} + \Delta C_{S_{r,r}} + \Delta C_{S_{e,l}} + \Delta C_{S_{e,r}} + \left[\left(\frac{\partial C_S}{\partial \beta} \right)_0 + \Delta \left(\frac{\partial C_S}{\partial \beta} \right)_{e,l} + \Delta \left(\frac{\partial C_S}{\partial \beta} \right)_{e,r} \right] \beta \tag{A.1b}$$

$$C_L = C_{L_0} + \Delta C_{L_{e,l}} + \Delta C_{L_{e,r}} + \Delta C_{L_b} \tag{A.1c}$$

$$C_l = \Delta C_{l_{e,l}} + \Delta C_{l_{e,r}} + \left(\frac{\partial C_l}{\partial \beta} \right)_0 \beta \tag{A.1d}$$

$$C_m = C_{m_0} + \Delta C_{l_{e,l}} + \Delta C_{l_{e,r}} + \Delta C_{l_b} \tag{A.1e}$$

$$C_n = \Delta C_{n_{r,l}} + \Delta C_{n_{r,r}} + \Delta C_{n_{e,l}} + \Delta C_{n_{e,r}} + \left[\left(\frac{\partial C_n}{\partial \beta} \right)_0 + \Delta \left(\frac{\partial C_n}{\partial \beta} \right)_{r,l} + \Delta \left(\frac{\partial C_n}{\partial \beta} \right)_{r,r} \right] \beta \tag{A.1f}$$

In Equations A.1, the aerodynamic force coefficients are the drag coefficient C_D, the side-force coefficient C_S , and the lift coefficient C_L, and the aerodynamic moment coefficients are the roll coefficient C_l, the pitch coefficient C_m, and the yaw coefficient C_n. The individual components and their dependencies are listed in Table A.1. Since the two rudders and elevons are symmetrically located with respect to the XZ-plane of the vehicle, there is a relation between their aerodynamic coefficients known as the symmetry condition. These relations are given in Table A.2.

Although the aerodynamic database of HORUS-2B is extensive, it is still only defined for certain trajectory points. The range of vehicle parameters for which the aerodynamic data is available is given in Table A.3. Values between these defined points can be found though interpolation, which is explained in Section 7.2.2. Outside of this region, extrapolation can be used. However, cases that do not lie within these constraints are almost nonexistent.

For more information on this database, such as the used assumptions and actual values of the coefficients, the reader is referred to Mooij (1995).

A

Table A.1: Aerodynamic force and moment coefficients functional arguments list (Mooij, 1995)

Drag coefficients	Side-force coefficients	Lift coefficients
$C_{D_0} = f(\alpha, M)$	$\Delta C_{S_{r,l}} = f(\alpha, \delta_{r,l}, M)$	$C_{L_0} = f(\alpha, M)$
$\Delta C_{D_{r,l}} = f(\alpha, \delta_{r,l}, M)$	$\Delta C_{S_{e,l}} = f(\alpha, \delta_{e,l}, M)$	$\Delta C_{L_{e,l}} = f(\alpha, \delta_{e,l}, M)$
$\Delta C_{D_{e,l}} = f(\alpha, \delta_{e,l}, M)$	$\Delta C_{S_{e,r}} = f(\alpha, \delta_{e,r}, M)$	$\Delta C_{L_b} = f(\alpha, \delta_b, M)$
$\Delta C_{D_b} = f(\alpha, \delta_b, M)$	$\Delta C_{S_{r,r}} = f(\alpha, \delta_{r,r}, M)$	$\Delta C_{L_{e,r}} = f(\alpha, \delta_{e,r}, M)$
$\Delta C_{D_{e,r}} = f(\alpha, \delta_{e,r}, M)$	$\left(\frac{\partial C_S}{\partial \beta}\right)_0 = f(\alpha, M)$	
$\Delta C_{D_{r,r}} = f(\alpha, \delta_{r,r}, M)$	$\Delta\left(\frac{\partial C_S}{\partial \beta}\right)_{e,l} = f(\alpha, \delta_{e,l}, M)$	
$\Delta C_{D_h} = f(\alpha, h)$	$\Delta\left(\frac{\partial C_S}{\partial \beta}\right)_{e,r} = f(\alpha, \delta_{e,r}, M)$	
Roll coefficients	**Pitch coefficients**	**Yaw coefficients**
$\Delta C_{l_{e,l}} = f(\alpha, \delta_{e,l}, M)$	$C_{m_0} = f(\alpha, M)$	$\Delta C_{n_{r,l}} = f(\alpha, \delta_{r,l}, M)$
$\Delta C_{l_{e,r}} = f(\alpha, \delta_{e,r}, M)$	$\Delta C_{m_{e,l}} = f(\alpha, \delta_{e,l}, M)$	$\Delta C_{n_{e,l}} = f(\alpha, \delta_{e,l}, M)$
$\left(\frac{\partial C_l}{\partial \beta}\right)_0 = f(\alpha, M)$	$\Delta C_{m_b} = f(\alpha, \delta_b, M)$	$\Delta C_{n_{e,r}} = f(\alpha, \delta_{e,r}, M)$
	$\Delta C_{m_{e,r}} = f(\alpha, \delta_{e,r}, M)$	$\Delta C_{n_{r,r}} = f(\alpha, \delta_{r,r}, M)$
		$\left(\frac{\partial C_n}{\partial \beta}\right)_0 = f(\alpha, M)$
		$\Delta\left(\frac{\partial C_n}{\partial \beta}\right)_{r,l} = f(\alpha, \delta_{r,l}, M)$
		$\Delta\left(\frac{\partial C_n}{\partial \beta}\right)_{r,r} = f(\alpha, \delta_{r,r}, M)$

Table A.2: Symmetry conditions for rudders and elevons (Mooij, 1995)

Zero derivatives elevon	First derivatives elevon	Zero derivatives rudder	First derivatives rudder
$C_{D_{e,r}} = C_{D_{e,l}}$	$\left(\frac{\partial C_S}{\partial \beta}\right)_{e,r} = \left(\frac{\partial C_S}{\partial \beta}\right)_{e,r}$	$C_{D_{r,r}} = C_{D_{r,l}}$	$\left(\frac{\partial C_n}{\partial \beta}\right)_{r,l} = \left(\frac{\partial C_n}{\partial \beta}\right)_{r,l}$
$C_{S_{e,r}} = -C_{S_{e,l}}$	$\left(\frac{\partial C_n}{\partial \beta}\right)_{e,r} = \left(\frac{\partial C_n}{\partial \beta}\right)_{e,r}$	$C_{S_{r,r}} = -C_{S_{r,l}}$	
$C_{L_{e,r}} = C_{L_{e,l}}$		$C_{n_{r,r}} = C_{n_{r,l}}$	
$C_{l_{e,r}} = -C_{l_{e,l}}$			
$C_{m_{e,r}} = C_{m_{e,l}}$			
$C_{n_{e,r}} = -C_{n_{e,l}}$			

Table A.3: Ranges of vehicle parameters and aerodynamic variables for the HORUS-2B aerodynamic database (Mooij, 1995).

α [°]	M [-]	δ_r [°]	δ_e [°]	δ_b [°]
0.0	1.2	0.0	-40.0	-20.0
5.0	1.5	10.0	-30.0	-10.0
10.0	2.0	20.0	-20.0	0.0
15.0	3.0	30.0	-10.0	10.0
20.0	5.0	40.0	0.0	20.0
25.0	10.0		10.0	30.0
30.0	20.0		20.0	
35.0			30.0	
40.0			40.0	
45.0				

B

FRAME TRANSFORMATIONS

Below all transformation matrices from Section 5.3, as well as graphical representations of the reference frames.

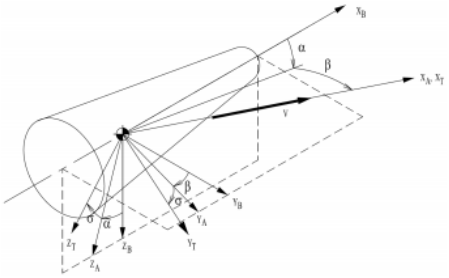


Figure B.1: Body-fixed (B), aerodynamic (A) and trajectory (T) reference frames (Mooij, 1994).

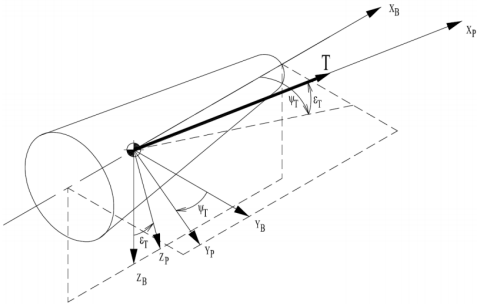


Figure B.2: Body-fixed (B) and propulsion (P) reference frames (Mooij, 1994).

$$\mathbf{C}_{A,B} = \mathbf{C}_Z(\beta)\mathbf{C}_Y(-\alpha) = \begin{bmatrix} \mathrm{c}\beta\mathrm{c}\alpha & \mathrm{s}\beta & \mathrm{c}\beta\mathrm{s}\alpha \\ -\mathrm{s}\beta\mathrm{c}\alpha & \mathrm{c}\beta & -\mathrm{s}\beta\mathrm{s}\alpha \\ -\mathrm{s}\alpha & 0 & \mathrm{c}\alpha \end{bmatrix} \tag{B.1}$$

$$\mathbf{C}_{T,A}(\sigma) = \mathbf{C}_X(\sigma) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \mathrm{c}\sigma & \mathrm{s}\sigma \\ 0 & -\mathrm{s}\sigma & \mathrm{c}\sigma \end{bmatrix} \tag{B.2}$$

$$\mathbf{C}_{P,B} = \mathbf{C}_Y(\epsilon_T)\mathbf{C}_Z(\psi_T) = \begin{bmatrix} \mathrm{c}\psi_T\mathrm{c}\epsilon_T & \mathrm{s}\psi_T\mathrm{c}\epsilon_T & -\mathrm{s}\epsilon_T \\ -\mathrm{s}\psi_T & \mathrm{c}\psi_T & 0 \\ \mathrm{c}\psi_T\mathrm{s}\epsilon_T & \mathrm{s}\psi_T\mathrm{s}\epsilon_T & \mathrm{c}\epsilon_T \end{bmatrix} \tag{B.3}$$

C

GIBBS ITERATION EQUATIONS

The reduced Gibbs iteration equations as given by McBride and Sanford (1994) can conveniently be organised in some sort of state-space form: **Ax** = **b**, with **x** the vector containing the correction variables as given in Section 3.1.3. The **A** and **b** matrices are given in Table C.1, where the 'right side' is the **b**-matrix.

S_j and μ_j in the equations are defined as follows:

$$S_j = S_j^0 - R\ln\frac{n_j}{n} - R\ln p \tag{C.1a}$$

$$\mu_j = \mu_j^0 + RT\ln\frac{n_j}{n} + RT\ln p \tag{C.1b}$$

As was explained in Section 3.1.4, some derivatives with respect to the logarithm of pressure or temperature are required to compute certain flow properties in the chamber and nozzle. These can be obtained in a similar way as the correction variables. An overview of the required relations is given in Tables C.2 and C.3.

Table C.1: Iteration equations to determine equilibrium compositions for the constant pressure, constant enthalpy (HP) and constant pressure, constant entropy (SP) problems (McBride and Sanford, 1994).

Problem	Variable						
	π_1	π_2	$\cdots$	π_v	$\Delta \ln n$	$\Delta \ln T$	Right side
HP/SP	$\sum_{j=1}^{NG} a_{1j} a_{1j} n_j$	$\sum_{j=1}^{NG} a_{1j} a_{2j} n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{1j} a_{vj} n_j$	$\sum_{j=1}^{NG} a_{1j} n_j$	$\sum_{j=1}^{NG} a_{1j} n_j \frac{H_j^0}{RT}$	$(b_1^0 - b_1) + \sum_{j=1}^{NG} a_{1j} n_j \frac{\mu_j}{RT}$
HP/SP	$\sum_{j=1}^{NG} a_{2j} a_{1j} n_j$	$\sum_{j=1}^{NG} a_{2j} a_{2j} n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{2j} a_{vj} n_j$	$\sum_{j=1}^{NG} a_{2j} n_j$	$\sum_{j=1}^{NG} a_{2j} n_j \frac{H_j^0}{RT}$	$(b_2^0 - b_2) + \sum_{j=1}^{NG} a_{2j} n_j \frac{\mu_j}{RT}$
HP/SP	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
HP/SP	$\sum_{j=1}^{NG} a_{vj} a_{1j} n_j$	$\sum_{j=1}^{NG} a_{vj} a_{2j} n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{vj} a_{vj} n_j$	$\sum_{j=1}^{NG} a_{vj} n_j$	$\sum_{j=1}^{NG} a_{vj} n_j \frac{H_j^0}{RT}$	$(b_v^0 - b_v) + \sum_{j=1}^{NG} a_{vj} n_j \frac{\mu_j}{RT}$
HP/SP	$\sum_{j=1}^{NG} a_{1j} n_j$	$\sum_{j=1}^{NG} a_{2j} n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{vj} n_j$	$\sum_{j=1}^{NG} n_j - n$	$\sum_{j=1}^{NG} n_j \frac{H_j^0}{RT}$	$n - \sum_{j=1}^{NG} n_j + \sum_{j=1}^{NG} n_j \frac{\mu_j}{RT}$
HP	$\sum_{j=1}^{NG} a_{1j} n_j \frac{H_j^0}{RT}$	$\sum_{j=1}^{NG} a_{2j} n_j \frac{H_j^0}{RT}$	$\cdots$	$\sum_{j=1}^{NG} a_{vj} n_j \frac{H_j^0}{RT}$	$\sum_{j=1}^{NG} n_j \frac{H_j^0}{RT}$	$\sum_{j=1}^{NS} n_j \frac{C_{p,j}^0}{R} + \sum_{j=1}^{NG} n_j \frac{H_j^0}{RT} \frac{H_j^0}{RT}$	$\frac{h_0}{RT} - \frac{h}{RT} + \sum_{j=1}^{NG} n_j \frac{H_j^0}{RT} \frac{\mu_j}{RT}$
SP	$\sum_{j=1}^{NG} a_{1j} n_j \frac{S_j}{R}$	$\sum_{j=1}^{NG} a_{2j} n_j \frac{S_j}{R}$	$\cdots$	$\sum_{j=1}^{NG} a_{vj} n_j \frac{S_j}{R}$	$\sum_{j=1}^{NG} n_j \frac{S_j}{R}$	$\sum_{j=1}^{NS} n_j \frac{C_{p,j}^0}{R} + \sum_{j=1}^{NG} n_j \frac{H_j^0}{RT} \frac{S_j}{R}$	$\frac{s_0}{R} - \frac{s}{R} + n - \sum_{j=1}^{NG} n_j + \sum_{j=1}^{NG} n_j \frac{S_j}{R} \frac{\mu_j}{RT}$

Table C.2: Equations for evaluating the derivatives with respect to the logarithm of temperature at constant pressure (McBride and Sanford, 1994).

Variable					
$\frac{\partial \pi_1}{\partial \ln T}$	$\frac{\partial \pi_2}{\partial \ln T}$	$\cdots$	$\frac{\partial \pi_v}{\partial \ln T}$	$\frac{\partial \ln n}{\partial \ln T}$	Right side
$\sum_{j=1}^{NG} a_{1j}a_{1j}n_j$	$\sum_{j=1}^{NG} a_{1j}a_{2j}n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{1j}a_{vj}n_j$	$\sum_{j=1}^{NG} a_{1j}n_j$	$-\sum_{j=1}^{NG} a_{1j}n_j \frac{H_j^0}{RT}$
$\sum_{j=1}^{NG} a_{2j}a_{1j}n_j$	$\sum_{j=1}^{NG} a_{2j}a_{2j}n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{2j}a_{vj}n_j$	$\sum_{j=1}^{NG} a_{2j}n_j$	$-\sum_{j=1}^{NG} a_{2j}n_j \frac{H_j^0}{RT}$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$
$\sum_{j=1}^{NG} a_{vj}a_{1j}n_j$	$\sum_{j=1}^{NG} a_{vj}a_{2j}n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{vj}a_{vj}n_j$	$\sum_{j=1}^{NG} a_{vj}n_j$	$-\sum_{j=1}^{NG} a_{vj}n_j \frac{H_j^0}{RT}$
$\sum_{j=1}^{NG} a_{1j}n_j$	$\sum_{j=1}^{NG} a_{2j}n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{vj}n_j$	0	$-\sum_{j=1}^{NG} n_j \frac{H_j^0}{RT}$

Table C.3: Equations for evaluating the derivatives with respect to the logarithm of pressure at constant temperature (McBride and Sanford, 1994).

Variable					
$\frac{\partial \pi_1}{\partial \ln p}$	$\frac{\partial \pi_2}{\partial \ln p}$	$\cdots$	$\frac{\partial \pi_v}{\partial \ln p}$	$\frac{\partial \ln n}{\partial \ln p}$	Right side
$\sum_{j=1}^{NG} a_{1j}a_{1j}n_j$	$\sum_{j=1}^{NG} a_{1j}a_{2j}n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{1j}a_{vj}n_j$	$\sum_{j=1}^{NG} a_{1j}n_j$	$\sum_{j=1}^{NG} a_{1j}n_j$
$\sum_{j=1}^{NG} a_{2j}a_{1j}n_j$	$\sum_{j=1}^{NG} a_{2j}a_{2j}n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{2j}a_{vj}n_j$	$\sum_{j=1}^{NG} a_{2j}n_j$	$\sum_{j=1}^{NG} a_{2j}n_j$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$
$\sum_{j=1}^{NG} a_{vj}a_{1j}n_j$	$\sum_{j=1}^{NG} a_{vj}a_{2j}n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{vj}a_{vj}n_j$	$\sum_{j=1}^{NG} a_{vj}n_j$	$\sum_{j=1}^{NG} a_{vj}n_j$
$\sum_{j=1}^{NG} a_{1j}n_j$	$\sum_{j=1}^{NG} a_{2j}n_j$	$\cdots$	$\sum_{j=1}^{NG} a_{vj}n_j$	0	$\sum_{j=1}^{NG} n_j$

D

BENCHMARK CONTROLLER

To verify the NDI controllers, a benchmark controller is designed. The Linear Quadratic Regulator (LQR) controller described by Mooij (1997) was chosen for this purpose. This section gives a concise overview the theory behind the LQR controller.

D.1. FEEDBACK CONTROL

The state-space system of the vehicle is described and derived in Appendix E. The open-loop system is usually not completely stable and is quite sensitive to disturbances. To mitigate these issues, feedback control can be introduced. This is also called closed-loop control, where a to-be-determined controller takes either the state or the output of a system (also called plant) as input and compares it with a command signal (Mooij, 2016). This system is graphically shown in Fig. D.1. An error between the output of the system and the commanded state is computed, which is then used to obtain a suitable control measure, to minimise or reduce this error. The EOM in state-space form and the output equations are given in Equations (E.5) and (E.7). If the state and input matrices **A** and **B** are constant over time, the system is called Linear Time Invariant (LTI) (Mooij, 2016). For the remainder of this section, the coefficient matrices are assumed to be constant and the system is considered to be LTI. This is a valid assumption as the system is linearised and the analysis is done over short time periods, so none of the coefficients in the matrices change. This because the coefficients only depend on parameters that are either constant for the whole mission, such as dimensional parameters, or depend on the reference conditions, which are constant as well per test case. Inertia and mass parameters are also assumed to be constant over the short analysis times.

The open-loop behaviour of the system can be obtained by calculating the eigenvalues and eigenvectors of the state matrix **A**. Usually, the open-loop results are not optimal which leads to the introduction of a feedback loop. This changes the eigenvalues of the system. The open-loop behaviour of a spaceplane was analysed in depth by Viavattene (2018). The closed-loop system can be tuned to make the system behaviour more desirable. The main principle of feedback control is thus to change the eigenvalues (and hence eigenmotion) of the open-loop system (Mooij, 2016).

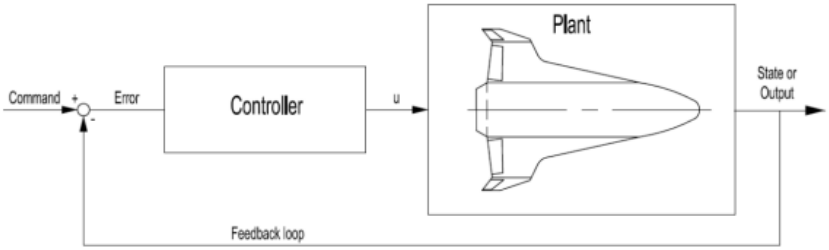


Figure D.1: Closed-loop control of an arbitrary plant by means of state or output feedback (Mooij, 2016)

The feedback-loop can either be state or output feedback. As explained in Appendix E, the output of the linear system is the state vector or a linear combination thereof. Hence, only state feedback is considered in this report, for which the control law is given as follows (Ogata, 2010):

$$\Delta \mathbf{u} = -\mathbf{K}\Delta \mathbf{x} \tag{D.1}$$

where $\mathbf{K}$ is the time independent feedback or gain matrix, which controls the behaviour of the closed-loop system.

The equation above can be substituted in Eq. (E.5), which yields:

$$\Delta \dot{\mathbf{x}} = (\mathbf{A} - \mathbf{BK})\Delta \mathbf{x} \tag{D.2}$$

The characteristic equation of Eq. (D.2), which gives us the eigenvalues and corresponding eigenmotion of the closed-loop system, is given by:

$$\det|\mathbf{A} - \mathbf{BK} - \lambda \mathbf{I}| = 0 \tag{D.3}$$

As becomes clear by studying the above equation, the gain is what eventually can be tuned to attain desired behaviour of the system (Mooij, 2016). By varying the gain matrix, the eigenvalues of the system can be changed, which is ultimately the purpose of feedback control. Of course, only varying the gain will not make any system controllable. The system itself should be controllable for this method to work. The gain matrix can be determined using optimal control theory (Mooij, 1998; Ogata, 2010), which is elaborated on in the following section.

D.2. OPTIMAL CONTROL THEORY

The state-space system was determined in the previous chapter as:

$$\Delta \dot{\mathbf{x}} = \mathbf{A}\Delta \mathbf{x} + \mathbf{B}\Delta \mathbf{u}, \quad \Delta \mathbf{y} = \mathbf{C}\Delta \mathbf{x} \tag{D.4}$$

For state feedback, the control law was defined to be (Mooij, 1998):

$$\Delta \mathbf{u} = -\mathbf{K}\Delta \mathbf{x} \tag{D.5}$$

The feedback matrix, **K** can be determined using Quadratic Optimal Control (Mooij, 1998), which defines a cost criterion that should be minimised. This cost criterion is defined in Eq. (D.6)

$$\mathbf{J} = \int_0^\infty \left(\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right) dt \tag{D.6}$$

where the first term in the integral represents the control deviation and the second term the control effort. The matrices **Q** and **R** can be varied to put more weight on the control deviation, resulting in a faster response, or the control effort, giving smaller control signals. These matrices are usually defined in an iterative manner to make the system react in a desirable way. As a start, the matrices can be chosen following *Bryson's Rule* (Mooij, 1998):

$$\mathbf{Q} = \text{diag}\left\{\frac{1}{\Delta x_{1_{max}}^2}, \frac{1}{\Delta x_{2_{max}}^2}, \cdots, \frac{1}{\Delta x_{n_{max}}^2}\right\} \tag{D.7a}$$

$$\mathbf{R} = \text{diag}\left\{\frac{1}{\Delta u_{1_{max}}^2}, \frac{1}{\Delta u_{2_{max}}^2}, \cdots, \frac{1}{\Delta u_{n_{max}}^2}\right\} \tag{D.7b}$$

with $\Delta x_{i_{max}}^2$ and $\Delta u_{j_{max}}^2$ the maximum allowable amplitude of the i-th element of the state vector and the j-th control, respectively.

Mooij (2016) describes the process to further determine the feedback matrix, which is not repeated here. Finally **K** is obtained from Eq. (D.8).

$$\mathbf{K} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P} \tag{D.8a}$$

$$\mathbf{0} = \mathbf{A}^T\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}\mathbf{P} + \mathbf{Q} \tag{D.8b}$$

(D.8b) is also known as the matrix Riccati equation. Solving this equation can commonly be done with standard algorithms, available in control-system design tools such as MATLAB.

D.3. BENCHMARK: OPTIMAL FEEDBACK CONTROL

At this point, it is possible to apply the optimal control theory from the previous chapter to HORUS. The notes by Mooij (1998) describe the control of the HORUS-2B vehicle, which is very similar to Sänger's HORUS vehicle.

As argued by Mooij (1998), the longitudinal and lateral motion of HORUS are not strongly coupled. Thus, the system can be decoupled to facilitate the coming analysis. To simplify the design of the controller, this fact is used to separate the controller in two parts: a longitudinal and lateral controller. For the LQR controller, as it is mainly a way of verification of the NDI controller, an additional simplification is made. It is assumed that the inertia matrix is a diagonal matrix. This is to make the controller more similar to the controllers implemented by Brinkman (2017) and Viavattene (2018).

D.3.1. LONGITUDINAL CONTROL

The longitudinal controller consists of an inner and outer loop. The inner loop takes care of stability augmentation while the outer loop provides the corrective control (Mooij, 2016). The inner loop solely relies on the aerodynamic surfaces (body flap and elevators) for trim, whereas the outer loop can combine both the aerodynamic surfaces as well as thruster jets, in case of descent, or the engines for the ascent phase.

Neglecting the asymmetric motion by setting $\Delta\beta = \Delta\sigma = 0°$, the following longitudinal equations of motions are given:

$$\Delta\dot{q} = \frac{1}{I_{yy}}\left(\frac{\partial C_m}{\partial\alpha}\Delta\alpha + \frac{\partial C_m}{\partial\delta_e}\Delta\delta_e\right)\bar{q}_e S_{ref} c_{ref} + \frac{\Delta M_{T_y}}{I_{yy}} \tag{D.9a}$$

$$\Delta\dot{\alpha} = \Delta q - \frac{1}{mV_e}\frac{\partial C_L}{\partial\alpha}\bar{q}_e S_{ref}\Delta\alpha \tag{D.9b}$$

with the symmetric elevator deflection and engine equivalence ratio as control variables. The elevator deflection is defined as:

$$\delta_e = \frac{\delta_{e,l} + \delta_{e,r}}{2} \tag{D.10}$$

Part of the aim of this project is to research how much and when the controls such as the elevators can be used effectively to control the vehicle. Aerodynamic surfaces are not very effective at high altitudes and are therefore usually only used above certain dynamic pressures. It is of interest to test at what points the controls are most effective.

The state-feedback laws for longitudinal control are selected to be simple proportional laws as argued by Mooij (1998):

$$\frac{\Delta\delta_e}{\delta_{e_{max}}} = -K_{eq}\Delta q - K_{e\alpha}\Delta\alpha \tag{D.11a}$$

$$\frac{\Delta M_{t_y}}{M_{T_{y,max}}} = -K_{yq}\Delta q - K_{y\alpha}\Delta\alpha \tag{D.11b}$$

where:

$\Delta q = q - q_e = q - q_c$
$\Delta\alpha = \alpha - \alpha_e \approx \alpha - \alpha_c$
q_c is the commanded pitch rate
α_c is the angle of attack from the guidance system (commanded angle of attack)

The above feedback laws are already a linear combination of the state variables and can easily be written in the form of Eq. (D.5):

$$\Delta\mathbf{u} = \begin{pmatrix}\Delta\delta_e \\ \Delta M_{T_y}\end{pmatrix} = -\begin{bmatrix}K_{eq} & K_{e\alpha} \\ K_{t_i q} & K_{t_i\alpha}\end{bmatrix}\begin{pmatrix}\Delta q \\ \Delta\alpha\end{pmatrix} \tag{D.12}$$

It should be noted that the actual **K** matrix is larger, but a shorter notation for the five engines is used here.

Due to the relation between q and α, thee above controller can be considered as a PD controller. Using the above equations, the Ricatti equation can be solved with weight matrices:

$$\mathbf{Q} = \text{diag}\left\{\frac{1}{\Delta q^2_{max}}, \frac{1}{\Delta \alpha^2_{max}}\right\} \quad , \quad \mathbf{R} = \text{diag}\left\{\frac{1}{\Delta \delta^2_{e_max}}, \frac{1}{\Delta M^2_{T_{y,max}}}\right\} \tag{D.13}$$

Corresponding values for the above parameters should still be selected based on the vehicle configuration and flight profile.

D.3.2. LATERAL CONTROL

Lateral control consists of only the outer, corrective loop, as only small moments are required to maintain stability. This control is achieved by using the ailerons, the rudders, the roll and yaw jets for descent, and ailerons, rudders and engines for ascent.

The lateral dynamics consist of a coupled motion of roll and yaw. Neglecting the symmetric motion by setting $\Delta\alpha = 0°$, the following lateral equations of motions are given for the lower stage:

$$\Delta\dot{p} = \frac{1}{I_{xx}}\left(\frac{\partial C_l}{\partial \beta}\Delta\beta + \frac{\partial C_l}{\partial \delta_a}\Delta\delta_a\right)\bar{q}_e S_{ref} b_{ref} + \frac{\Delta M_{T_x}}{I_{xx}} \tag{D.14a}$$

$$\Delta\dot{r} = \frac{1}{I_{zz}}\left(\frac{\partial C_n}{\partial \beta}\Delta\beta + \frac{\partial C_n}{\partial \delta_a}\Delta\delta_a + \frac{\partial C_n}{\partial \delta_r} Delta\delta_r\right)\bar{q}_e S_{ref} b_{ref} + \frac{\Delta M_{T_z}}{I_{zz}} \tag{D.14b}$$

$$\Delta\dot{\beta} = \sin\alpha_e \Delta p - \cos\alpha_e \Delta r - \frac{1}{mV_0}\frac{\partial C_S}{\partial \beta}\bar{q}_e S_{ref}\Delta\beta - \frac{g_0}{V_0}\cos\gamma_e \sin\sigma_e \Delta\sigma \tag{D.14c}$$

$$\begin{aligned}\Delta\dot{\sigma} = &-\cos\alpha_e \Delta p - \sin\alpha_e \Delta r - \left(\frac{L_e}{mV_e} - \frac{g_e}{V_e}\cos\gamma_e\cos\sigma_e + \right. \\ &\left. + \frac{\tan\gamma_e}{mV_e}\cos\sigma_e \frac{\partial C_S}{\partial \beta}\bar{q}_e S_{ref}\right)\Delta\beta + \frac{\tan\gamma_e}{mV_e}\cos\sigma_e L_e \Delta\sigma\end{aligned} \tag{D.14d}$$

The aileron deflection is given in Eq. (D.15). The rudder deflection angle δ_r is defined to be equal to δ_r, when $\delta_{r,l}$ is positive, and equal to $-\delta_{r,r}$ when $\delta_{r,r}$ is positive (Mooij, 1998).

$$\delta_a = \frac{\delta_{e,l} - \delta_{e,r}}{2} \tag{D.15}$$

The control laws for aileron and rudder are given below:

$$\frac{\Delta\delta_a}{\delta_{a_{max}}} = -\left[K_{ap}\dot{\sigma} + K_{a\sigma}(\sigma - \sigma_c)\right]\cos\alpha - \left[K_{ar}(\dot{\beta} + \frac{g}{V}\sin\sigma) + K_{a\beta}\beta\right]\sin\alpha \tag{D.16a}$$

$$\frac{\Delta\delta_r}{\delta_{r_{max}}} = -\left[K_{rp}\dot{\sigma} + K_{r\sigma}(\sigma - \sigma_c)\right]\cos\alpha - \left[K_{rr}(\dot{\beta} + \frac{g}{V}\sin\sigma) + K_{r\beta}\beta\right]\sin\alpha \tag{D.16b}$$

where:

$$\Delta\beta = \beta - \beta_e = \beta - \beta_c = \beta$$
$$\Delta\sigma = \sigma - \sigma_e \approx \sigma - \sigma_c$$
$$\Delta p = p - p_e = p - p_c$$
$$\Delta r = r - r_e = r - r_c$$

The laws for jet controls are the same as for the longitudinal control and are not repeated here. The above control laws can be linearised as done by Mooij (1998):

$$\Delta\mathbf{u} = \begin{pmatrix}\Delta\delta_a\\ \Delta\delta_r\end{pmatrix} = -\begin{bmatrix}\delta_{a_{max}}K^*_{ap} & \delta_{a_{max}}K^*_{ar} & \delta_{a_{max}}K^*_{a\beta} & \delta_{a_{max}}K^*_{a\sigma}\\ \delta_{r_{max}}K^*_{rp} & \delta_{r_{max}}K^*_{rr} & \delta_{r_{max}}K^*_{r\beta} & \delta_{r_{max}}K^*_{r\sigma}\end{bmatrix}\begin{pmatrix}\Delta p\\ \Delta r\\ \Delta\beta\\ \Delta\sigma\end{pmatrix} \tag{D.17}$$

with:

$$\begin{pmatrix}K^*_{ap}\\ K^*_{a\sigma}\\ K^*_{ar}\\ K^*_{a\beta}\end{pmatrix} = -\begin{bmatrix}-\cos^2\alpha_e & 0 & \sin^2\alpha_e & 0\\ -\sin\alpha_e\cos\alpha_e & 0 & -\sin\alpha_e\cos\alpha_e & 0\\ 0 & 0 & 0 & \sin\alpha_e\\ 0 & \cos\alpha_e & \frac{g_e}{V_e}\cos\sigma_e\cos\alpha_e & 0\end{bmatrix}\begin{pmatrix}K_{ap}\\ K_{a\sigma}\\ K_{ar}\\ K_{a\beta}\end{pmatrix} \tag{D.18a}$$

$$\begin{pmatrix}K^*_{rp}\\ K^*_{r\sigma}\\ K^*_{rr}\\ K^*_{r\beta}\end{pmatrix} = -\begin{bmatrix}-\sin\alpha_e\cos\alpha_e & 0 & \sin\alpha_e\cos\alpha_e & 0\\ -\sin^2\alpha_e & 0 & -\cos^2\alpha_e & 0\\ 0 & 0 & 0 & \cos\alpha_e\\ 0 & \sin\alpha_e & \frac{g_e}{V_e}\cos\sigma_e\cos\alpha_e & 0\end{bmatrix}\begin{pmatrix}K_{rp}\\ K_{r\sigma}\\ K_{rr}\\ K_{r\beta}\end{pmatrix} \tag{D.18b}$$

D.4. BENCHMARK: LQR WITH MOMENT FRACTIONS

The overall controller using moment fractions is similar to the one described in the previous sections. The effectiveness of this controller lies in the fact that moment fractions are general parameters, which are independent of the actual active actuators. This controller thus directly commands the required moments about each axis, instead of the individual actuators. this makes it more simple to apply this controller to different flight conditions and situations.

The linearised equations for the kinematics:$\Delta\dot{\alpha}$, $\Delta\dot{\beta}$ and $\Delta\dot{\sigma}$ remain the same as before. However, the dynamic equations slightly change as follows:

$$\Delta\dot{p} = \frac{\bar{q}_e S_{ref} b_{ref}}{I_{xx}}\frac{\partial C_l}{\partial\beta}\Delta\beta + \frac{\mathscr{L}_{max}}{I_{xx}}\eta_x$$
$$\Delta\dot{q} = \frac{\bar{q}_e S_{ref} c_{ref}}{I_{yy}}\frac{\partial C_m}{\partial\alpha}\Delta\alpha + \frac{\mathscr{M}_{max}}{I_{yy}}\eta_y \tag{D.19a}$$
$$\Delta\dot{r} = \frac{\bar{q}_e S_{ref} b_{ref}}{I_{zz}}\frac{\partial C_n}{\partial\beta}\Delta\beta + \frac{\mathscr{N}_{max}}{I_{zz}}\eta_x \tag{D.19b}$$

with $\mathscr{L}_{max}$, $\mathscr{M}_{max}$ and $\mathscr{N}_{max}$ the maximum available moments, which are elaborated on in Section 6.4.

The longitudinal control law with moment fractions is:

$$\eta_y = -K_{yq}\Delta q - K_{y\alpha}\Delta\alpha \tag{D.20}$$

and the lateral control laws are:

$$\Delta\mathbf{u} = \begin{pmatrix}\eta_x\\ \eta_y\end{pmatrix} = -\begin{bmatrix}K_{xp} & K_{xr} & K_{x\beta} & K_{x\sigma}\\ K_{zp} & K_{zr} & K_{z\beta} & K_{z\sigma}\end{bmatrix}\begin{pmatrix}\Delta p\\ \Delta r\\ \Delta\beta\\ \Delta\sigma\end{pmatrix} = \mathbf{K}\Delta\mathbf{x} \tag{D.21}$$

The weighting matrices **Q** are the same as before, and the weighting matrices **R** for longitudinal and lateral control are:

$$\mathbf{R} = \frac{1}{\eta_{y,max}^2} \quad , \quad \mathbf{R} = \text{diag}\left\{\frac{1}{\eta_{x,max}^2}, \frac{1}{\eta_{y,max}^2}\right\} \tag{D.22}$$

where the maximum allowed amplitude of the control variables are $\eta_{x,max} = \eta_{y,max} = \eta_{z,max} = 1$.

E

Linearised Equations & State-Space Representation

For the LQR controller, the state-space system of the vehicle is required. Before the state-space can be determined, the linearised equations must be derived. Note that the state-space and linearised equations are only developed for and used by the LQR controller. These do not relate to the NDI controller or any other part of this study.

E.1. Linearised Equations

For the benchmark controller LQR controller described in Appendix D, a state-space representation of the system is required. A state-space system is also interesting to study both the open- and closed-loop response and eigenmotion of the system, and to analyse its stability. Equations (5.17) and (5.22) should be linearised before a state-space system can be formed. The process of linearisation is not explained in this study. The reader is referred to Mooij (2016); Mulder et al. (2013) for a more detailed explanation on the subject. Two examples are given in Appendix B. The linearised equations for the kinematic relations are taken from Mooij (2016); Mulder et al. (2013), as they have derived these equations properly. The linearised equations around a flight condition, considering a single point in the trajectory ($\Delta V = \Delta h = \Delta R = \beta_0 = S_0 = 0$ for example), denoted by the subscript '0' are given as:

$$\Delta\dot{p} = \frac{1}{I_{xx}}\Delta M_x \tag{E.1a}$$

$$\Delta\dot{q} = \frac{1}{I_{yy}}\Delta M_y \tag{E.1b}$$

$$\Delta\dot{r} = \frac{1}{I_{zz}}\Delta M_z \tag{E.1c}$$

$$\Delta\dot{\alpha} = \Delta q - \frac{1}{mV_0}\Delta L - \frac{g_0}{V_0}\cos\gamma_0 \sin\sigma_0 \Delta\sigma \tag{E.2a}$$

$$\Delta\dot{\beta} = \sin\alpha_0 \Delta p - \cos\alpha_0 \Delta r - \frac{\Delta S}{mV_0} - \frac{g_0}{V_0}\cos\gamma_0 \sin\sigma_0 \Delta\sigma \tag{E.2b}$$

$$\Delta\dot{\sigma} = -\cos\alpha_0 \Delta p - \sin\alpha_0 \Delta r - \left(\frac{L_0}{mV_0} - \frac{g_0}{V_0}\cos\gamma_0 \cos\sigma_0\right)\Delta\beta + \tag{E.2c}$$
$$+ \frac{\tan\gamma_0}{mV_0}\left(\sin\sigma_0 \Delta L + \cos\sigma_0 L_0 \Delta\sigma + \cos\sigma_0 \Delta S\right)$$

with:

$$\Delta L = \frac{\partial C_L}{\partial\alpha}\bar{q}_0 S_{ref}\Delta\alpha \tag{E.3a}$$

$$\Delta S = \frac{\partial C_S}{\partial\beta}\bar{q}_0 S_{ref}\Delta\beta \tag{E.3b}$$

and:

$$\Delta M_x = \frac{\partial C_l}{\partial\beta}\bar{q}_0 S_{ref} b_{ref}\Delta\beta + \frac{\partial C_l}{\partial\delta_a}\bar{q}_0 S_{ref} b_{ref}\Delta\delta_a + \Delta M_{T_x} \tag{E.4a}$$

$$\Delta M_y = \frac{\partial C_m}{\partial\alpha}\bar{q}_0 S_{ref} c_{ref}\Delta\alpha + \frac{\partial C_m}{\partial\delta_e}\bar{q}_0 S_{ref} c_{ref}\Delta\delta_e + \Delta M_{T_y} \tag{E.4b}$$

$$\Delta M_z = \frac{\partial C_n}{\partial\beta}\bar{q}_0 S_{ref} b_{ref}\Delta\beta + \frac{\partial C_n}{\partial\delta_r}\bar{q}_0 S_{ref} b_{ref}\Delta\delta_r + \frac{\partial C_n}{\partial\delta_a}\bar{q}_0 S_{ref} b_{ref}\Delta\delta_a + \tag{E.4c}$$
$$+ \Delta M_{T_z}$$

In the above equations, the partial derivatives can be obtained from the aerodynamic and propulsion models. The ranges for which these derivatives are available are given in Section 2.3.2. All the aerodynamic coefficients are available at certain trajectory points. Interpolation, as described in Section 7.2.2, is used to obtain the coefficients at intermediate points where no exact value is known. For the LQR, the inertia matrix is assumed to be a diagonal matrix, with all off-diagonal elements equal to zero (Mooij, 1998).

E.2. STATE-SPACE REPRESENTATION

The analysis of a vehicle's motion can greatly be simplified through linearisation of its equations of motion. The nonlinear equations can be linearised around a certain flight condition to make first-order predictions of the future states of the vehicle. The linearised equations can be put in a state-space formulation in matrix form, as given in Eq. (E.5), where $\Delta\mathbf{x}$ is the $n \times 1$ state vector, $\Delta\mathbf{u}$ is the $m \times 1$ input vector, and $\mathbf{A}$ and $\mathbf{B}$ are the $n \times n$ state matrix and the $n \times m$ input matrix, respectively (Mooij, 2016).

$$\Delta\dot{\mathbf{x}} = \mathbf{A}\Delta\mathbf{x} + \mathbf{B}\Delta\mathbf{u} \tag{E.5}$$

The state vector is defined as:

$$\Delta\mathbf{x} = \left(\Delta p, \Delta q, \Delta r, \Delta\alpha, \Delta\beta, \Delta\sigma\right)^T \tag{E.6}$$

The system can be completed by adding the output equation:

$$\Delta\mathbf{y} = \mathbf{C}\Delta\mathbf{x} + \mathbf{D}\Delta\mathbf{u} \tag{E.7}$$

where $\Delta\mathbf{y}$ is the $k \times 1$ output vector, $\mathbf{C}$ and $\mathbf{D}$ are the $k \times n$ output and $k \times m$ transmission matrices, respectively (Ogata, 2010), and $\Delta\mathbf{u}$ is the control vector given by:

$$\Delta\mathbf{u} = \left(\Delta\delta_e, \Delta\delta_a, \Delta\delta_r, \Delta M_{T,x}, \Delta M_{T,y}, \Delta M_{T,z}\right)^T \tag{E.8}$$

For this study, the output can be taken as the state itself of a linear combination thereof, as argued by Mooij (2016). Hence, the matrix $\mathbf{D}$ is a zero matrix and can therefore be excluded, leading to the new output equation:

$$\Delta\mathbf{y} = \mathbf{C}\Delta\mathbf{x} \tag{E.9}$$

The full state-space system for HORUS is given as:

$$\begin{pmatrix} \Delta\dot{p} \\ \Delta\dot{q} \\ \Delta\dot{r} \\ \Delta\dot{\alpha} \\ \Delta\dot{\beta} \\ \Delta\dot{\sigma} \end{pmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & a_{p\beta} & 0 \\ 0 & 0 & 0 & a_{q\alpha} & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{r\beta} & 0 \\ 0 & a_{\alpha q} & 0 & a_{\alpha\alpha} & 0 & a_{\alpha\sigma} \\ a_{\beta p} & 0 & a_{\beta r} & 0 & a_{\beta\beta} & a_{\beta\sigma} \\ a_{\sigma p} & 0 & a_{\sigma r} & a_{\sigma\alpha} & a_{\sigma\beta} & a_{\sigma\sigma} \end{bmatrix} \begin{pmatrix} \Delta p \\ \Delta q \\ \Delta r \\ \Delta\alpha \\ \Delta\beta \\ \Delta\sigma \end{pmatrix} + \begin{bmatrix} 0 & b_{pa} & b_{pr} & b_{pt_x} & 0 & 0 \\ b_{qe} & 0 & 0 & 0 & b_{qt_y} & 0 \\ 0 & b_{ra} & b_{rr} & 0 & 0 & b_{rt_z} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} \Delta\delta_e \\ \Delta\delta_a \\ \Delta\delta_r \\ \Delta M_{T,x} \\ \Delta M_{T,y} \\ \Delta M_{T,z} \end{pmatrix} \tag{E.10}$$

Each of the components a_{ij}, b_{ij} can be determined from the linearised equations from the previous section. The components are given in Table E.1.

The previous system relies on a set of actuators that vary throughout the flight of the vehicle. At high altitudes, the aerodynamic control surfaces are not effective and the control is completely carried out by the engines during the ascent phase, and the jets during descent. If the controls vary, the state-space system must be updated accordingly, which is time-consuming and rather impractical. Therefore, it is interesting to use control parameters as input, which are independent of the flight conditions. These are the so-called *moment fractions*, which are defined as the fraction of the maximum available control moments. Afterwards, these general parameters can be translated to the actual control inputs (surface deflections, thrust moments) by the actuator assignment algorithms that are given in Section 6.4.

The moment fractions are defined as:

$$\mathbf{u} = (\eta_x, \eta_y, \eta_z)^T = \left(\frac{\mathscr{L}_{cmd}}{\mathscr{L}_{max}}, \frac{\mathscr{M}_{cmd}}{\mathscr{M}_{max}}, \frac{\mathscr{N}_{cmd}}{\mathscr{N}_{max}} \right) \tag{E.11}$$

The state-space model in terms of the moment fractions is given as follows:

$$\begin{pmatrix} \Delta\dot{p} \\ \Delta\dot{q} \\ \Delta\dot{r} \\ \Delta\dot{\alpha} \\ \Delta\dot{\beta} \\ \Delta\dot{\sigma} \end{pmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & a_{p\beta} & 0 \\ 0 & 0 & 0 & a_{q\alpha} & 0 & 0 \\ 0 & 0 & 0 & 0 & a_{r\beta} & 0 \\ 0 & a_{\alpha q} & 0 & a_{\alpha\alpha} & 0 & a_{\alpha\sigma} \\ a_{\beta p} & 0 & a_{\beta r} & 0 & a_{\beta\beta} & a_{\beta\sigma} \\ a_{\sigma p} & 0 & a_{\sigma r} & a_{\sigma\alpha} & a_{\sigma\beta} & a_{\sigma\sigma} \end{bmatrix} \begin{pmatrix} \Delta p \\ \Delta q \\ \Delta r \\ \Delta\alpha \\ \Delta\beta \\ \Delta\sigma \end{pmatrix} + \begin{bmatrix} b_{p\eta_x} & 0 & 0 \\ 0 & b_{q\eta_y} & 0 \\ 0 & 0 & b_{r\eta_z} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} \eta_x \\ \eta_y \\ \eta_z \end{pmatrix} \tag{E.12}$$

where **A** is the same as for the previous controller and with b_{ij} given as:

$$b_{p\eta_x} = \frac{\mathscr{L}_{max}}{I_{xx}}$$
$$b_{q\eta_y} = \frac{\mathscr{M}_{max}}{I_{yy}}$$
$$b_{r\eta_z} = \frac{\mathscr{N}_{max}}{I_{zz}}$$

and all coefficients given in Table E.1.

Table E.1: Iteration equations to determine equilibrium compositions for the constant pressure, constant enthalpy (HP) and constant pressure, constant entropy (SP) problems (McBride and Sanford, 1994).

	p	q	r	α	β	σ
p	$a_{pp} = 0$	$a_{pq} = 0$	$a_{pr} = 0$	$a_{p\alpha} = 0$	$a_{p\beta} = \frac{1}{I_{xx}} \frac{\partial C_l}{\partial \beta} \bar{q}_0 S_{ref} b_{ref}$	$a_{p\sigma} = 0$
q	$a_{qp} = 0$	$a_{qq} = 0$	$a_{qr} = 0$	$a_{q\alpha} = \frac{1}{I_{yy}} \frac{\partial C_m}{\partial \alpha} \bar{q}_0 S_{ref} c_{ref}$	$a_{q\beta} = 0$	$a_{q\sigma} = 0$
r	$a_{rp} = 0$	$a_{rq} = 0$	$a_{rr} = 0$	$a_{r\alpha} = 0$	$a_{r\beta} = \frac{1}{I_{zz}} \frac{\partial C_n}{\partial \beta} \bar{q}_0 S_{ref} b_{ref}$	$a_{r\sigma} = 0$
α	$a_{\alpha p} = 0$	$a_{\alpha q} = 1$	$a_{\alpha r} = 0$	$a_{\alpha\alpha} = -\frac{1}{mV_0} \frac{\partial C_L}{\partial \alpha} \bar{q}_0 S_{ref}$	$a_{\alpha\beta} = 0$	$a_{\alpha\sigma} = -\frac{g_0}{V_0} \cos\gamma_0 \sin\sigma_0$
β	$a_{\beta p} = \sin\alpha_0$	$a_{\beta q} = 0$	$a_{\beta r} = -\cos\alpha_0$	$a_{\beta\alpha} = 0$	$a_{\beta\beta} = -\frac{1}{mV_0} \frac{\partial C_S}{\partial \beta} \bar{q}_0 S_{ref}$	$a_{\beta\sigma} = -\frac{g_0}{V_0} \cos\gamma_0 \sin\sigma_0$
σ	$a_{\sigma p} = -\cos\alpha_0$	$a_{\sigma q} = 0$	$a_{\sigma r} = -\sin\alpha_0$	$a_{\sigma\alpha} = \frac{\tan\gamma_0}{mV_0} \sin\sigma_0 \frac{\partial C_L}{\partial \alpha} \bar{q}_0 S_{ref}$	$a_{\sigma\beta} = -\frac{L_0}{mV_0} + \frac{g_0}{V_0} \cos\gamma_0 \cos\sigma_0 + + \frac{\tan\gamma_0}{mV_0} \cos\sigma_0 \frac{\partial C_S}{\partial \beta} \bar{q}_0 S_{ref}$	$a_{\sigma\sigma} = \frac{\tan\gamma_0}{mV_0} \cos\sigma_0 L_0$
	$\Delta M_{T,x}$	$\Delta M_{T,y}$	$\Delta M_{T,z}$	$\Delta\delta_e$	$\Delta\delta_a$	$\Delta\delta_r$
p	$b_{pt_x} = \frac{1}{I_x x}$	$b_{pt_y} = 0$	$b_{pt_z} = 0$	$b_{pe} = 0$	$b_{pa} = \frac{1}{I_{xx}} \frac{\partial C_l}{\partial \delta_a} \bar{q}_0 S_{ref} b_{ref}$	$b_{pr} = 0$
q	$b_{qt_x} = 0$	$b_{qt_y} = \frac{1}{I_{yy}}$	$b_{qt_z} = 0$	$b_{qe} = \frac{1}{I_{yy}} \frac{\partial C_m}{\partial \delta_e} \bar{q}_0 S_{ref} c_{ref}$	$b_{qa} = 0$	$b_{qr} = 0$
r	$b_{rt_x} = 0$	$b_{rt_y} = 0$	$b_{rt_z} = \frac{1}{I_{zz}}$	$b_{re} = 0$	$b_{ra} = \frac{1}{I_{zz}} \frac{\partial C_n}{\partial \delta_a} \bar{q}_0 S_{ref} b_{ref}$	$b_{rr} = \frac{1}{I_{zz}} \frac{\partial C_n}{\partial \delta_r} \bar{q}_0 S_{ref} b_{ref}$
α	$b_{\alpha t_x} = 0$	$b_{\alpha t_y} = 0$	$b_{\alpha t_z} = 0$	$b_{\alpha e} = 0$	$b_{\alpha a} = 0$	$b_{\alpha r} = 0$
β	$b_{\beta t_x} = 0$	$b_{\beta t_y} = 0$	$b_{\beta t_z} = 0$	$b_{\beta e} = 0$	$b_{\beta a} = 0$	$b_{\beta r} = 0$
σ	$b_{\sigma t_x} = 0$	$b_{\sigma t_y} = 0$	$b_{\sigma t_z} = 0$	$b_{\sigma e} = 0$	$b_{\sigma a} = 0$	$b_{\sigma r} = 0$