Autonoumous Vehicle Control Systems

Robust Flight Control Assignment

School of Aerospace, Transport and Manufacturing
Autonomous Vehicle Dynamics and Control, Msc.
Module: Autonomous Vehicle Control
Cranfield University
Cranfield, United Kingdom

Johannes Autenrieb
April 2019

Contents

List of Figures

List of Tables

Chapter 1

Flight Dynamics

1.1 Coordindate Systems

In order to describe parametric influences on the flight vehicle in space Different coordinate systems are applied. Since there are different fields of utilisation for a coordinate system, they differ from each other depending on the purpose and the value they would like to display. In this part, appropriate coordinate systems for the assigned problem will be presented.

1.1.1 Earth-Fixed Coordindate System

The earth-fixed coordindate system describes the global motion of an entity relativeto the earth. The three axes are used to represent the aircraft location. A common assumption is the flat earth assumption in near earth flight.[3; 8]

- x-axis: x_E or x_E
- y-axis: y_E or x_E s
- z-axis: : z_E or x_E

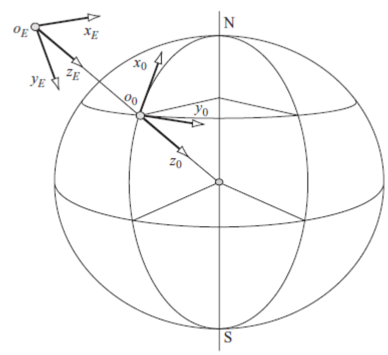

Figure 1.1: The Earth-fixed frame

1.1.2 Body-Axes

The coordinate systems origing is the CG of the missle. x_b has its direction into the way of the fuselage nose, and the z_b is pointing downwards.[3; 8]

- x-axis: x_b or B_1
- y-axis: y_b or B_2
- z-axis: : z_b or B_3

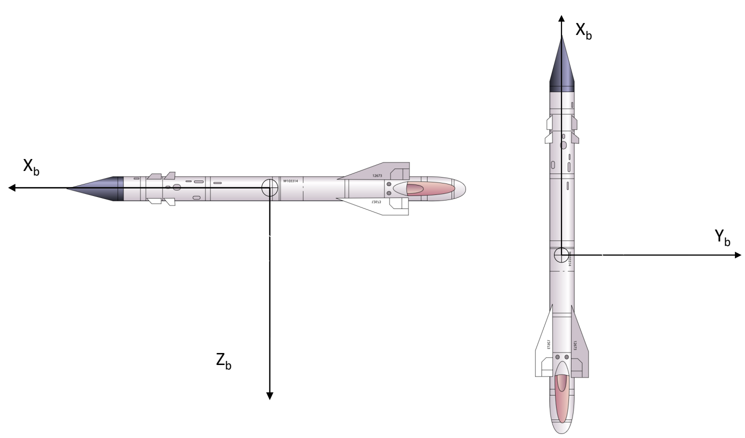

Figure 1.2: The body coordinate system

1.1.3 Aerodynamic Axes or Stability Axes

The direction of the airstream vector determines the direction of the aerodynamic coordinate system. Like the bod frame system is the origin of this frame also in the CG of the missile. The x_a is alligned with the airstream, while the others are orthogonally defined. [3; 8]

- x-axis: x_a or W_1
- y-axis: y_a or W_2
- z-axis: : z_a or W_3

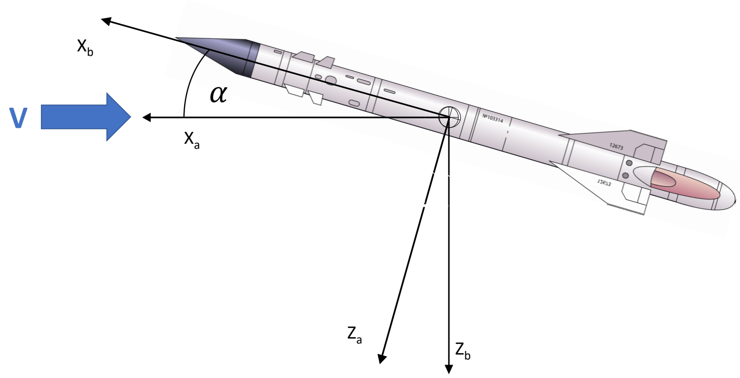

Figure 1.3: The aerodynamic coordinate system

1.2 Euler Angles

The Euler angles are utilised to formulate the orientation of an aircraft corresponding to the earth-fixed coordinate system. [3]

- Roll Angle: ϕ
- Pitch Angle: θ
- Yaw Angle: : ψ

1.3 Equation of Motion

Newtons law allows it to describe the rotational and translatory motion of a moving vehicle.[6; 2]

$$\sum \overrightarrow{F} = m\vec{\dot{v}} = \frac{d(m\vec{v})}{dt} \tag{1.1}$$

$$\sum \overrightarrow{M} = \frac{d}{dt}(H) \tag{1.2}$$

That is driving to the force and moment components arond three axes:

$$\begin{array}{ccc} F_X = \frac{d(mu)}{dt} & F_Y = \frac{d(mv)}{dt} & F_Z = \frac{d(mw)}{dt} \\ L = \frac{d}{dt}H_X & M = \frac{d(m\vec{v})}{dt}H_Y & N = \frac{d(m\vec{v})}{dt}H_Z \end{array} \tag{1.3}$$

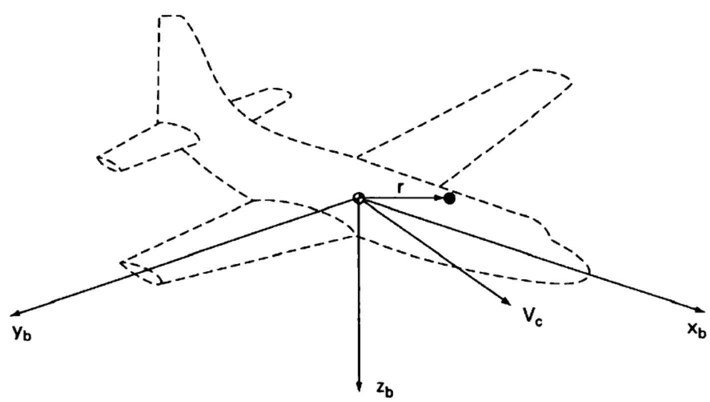

Figure 1.4: Element on an arbitrary Aircraft[6]

The forces represented in the body-fixed coordinate system for a small mass are:

$$\delta F = \delta m \frac{d(v)}{dt} \tag{1.4}$$

$$\delta F = F \tag{1.5}$$

The mass velocity would b defined as:

$$v_P = v_c + \frac{dr}{dt} \tag{1.6}$$

$$\sum \delta F = \frac{d}{dt}(v_c + \frac{dr}{dt})\delta m \tag{1.7}$$

With a assuming that the velocity is constant, we receive:

$$F = m\frac{d}{dt}v_c + \frac{d}{dt}\sum \frac{dr}{dt}\delta m \tag{1.8}$$

$$F = m\frac{d}{dt}v_c + \frac{d^2}{dt^2}\sum r\,\delta m \tag{1.9}$$

Because $\sum r\,\delta m$ is zero, we get:

$$F = m\frac{d}{dt}v_c \tag{1.10}$$

This methodology can be used to receive the rotatory equations.

$$\delta M = \frac{d}{dt}\delta H = \frac{d}{dt}(r \times v)\delta m \tag{1.11}$$

The angular rate or velocity of a mass is:

$$v_P = v_c + \frac{dr}{dt} = \omega \times r \tag{1.12}$$

Sum of all moments is:

$$H = \delta H = \sum(r \times v_c)\delta m + \sum[r \times (\omega \times r)]\delta m \tag{1.13}$$

With assuming that the angular velocity is not changing over the time, than we receive:

$$H = \sum r\,\delta m \times v_c + \sum[r \times (\omega \times r)]\delta m \tag{1.14}$$

The fact that $\sum r\,\delta m$ is zero, we get:s:

$$\omega = pi + qj + rk \tag{1.15}$$

$$r = xi + yj + zk \tag{1.16}$$

By substitituing the velocities as single components in to the prior equatings we receive:

$$H = (pi + qj + rk)\sum(x^2 + y^2 + z^2)\,\delta m - \sum(xi + yj + zk)(pi + qj + rk)\delta m \tag{1.17}$$

This leads to following components::

$$H_X = p\sum(y^2 + z^2)\,\delta m - \sum xy\,\delta m - \sum r\,\delta m \tag{1.18}$$

$$H_Y = -p\sum xy\,\delta m + \sum q(x^2 + y^2)\delta m - r\sum yz\,\delta m \tag{1.19}$$

$$H_Z = -p\sum xz\,\delta m - q\sum yz\,\delta m + r\sum(x^2 + y^2)\,\delta m \tag{1.20}$$

Summation of all finite mass elements delivers:.

$$\begin{array}{ll} I_x = \int\int\int \quad (y^2 + z^2)\,\delta m & I_{xy} = \int\int\int xy\,\delta m \\ I_y = \int\int\int \quad (x^2 + z^2)\,\delta m & I_{xz} = \int\int\int xz\,\delta m \\ I_z = \int\int\int \quad (x^2 + y^2)\,\delta m & I_{yz} = \int\int\int yz\,\delta m \end{array} \tag{1.21}$$

This leads to follwing equation:

$$H_x = pI_x - qI_{xy} - rI_{xz} \tag{1.22}$$

$$H_y = -pI_{xy} + qI_y - rI_{yz} \tag{1.23}$$

$$H_z = -pI_{xz} - qI_{xy} + rI_z \tag{1.24}$$

We finally can obtain the equations for the linear motion:

$$F_X = m(\dot{u} + qw - rv) \tag{1.25}$$

$$F_Y = m(\dot{v} + ru - pw) \tag{1.26}$$

$$F_Z = m(\dot{w} + pv - qu) \tag{1.27}$$

And for the translatory motion:

$$L = \dot{H}_X + qH_Z - rH_Y = I_x\dot{p} - I_x\dot{r} + qr(I_z - I_y) - I_{xz}pq \tag{1.28}$$

$$M = \dot{H}_Y + rH_X - pH_Z = I_y\dot{q} + rp(I_x - I_z) + I_{xz}(p^2 - r^2) \tag{1.29}$$

$$N = \dot{H}_Z + pH_Y - qH_X = -I_{xz}\dot{p} + I_z\dot{r} + pq(I_y - I_x) + I_{xz}qr \tag{1.30}$$

1.4 Moment of Intertia Tensor

The tensor for the moment of inertia can be used to describe rotational intertia of a missile.[6; 3]

$$\underline{\underline{I}} = \begin{pmatrix} I_x & I_{xy} & I_{xz} \\ I_{yx} & I_y & I_x \\ X_{zx} & I_{yz} & I_z \end{pmatrix} \tag{1.31}$$

For cases of symmetries some elemenmts of the tensor getting zero. For the case of the missile we have symmetries around the y and z directions This leads diagonal matrixe:

$$\underline{\underline{I}} = \begin{pmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{pmatrix} \tag{1.32}$$

1.5 Aerodynamics

In order to describe the aerodynamic actions which are created by small perturbations, it is needed to define the right equations. The pertubation allows us to assume that a change of value can be traced back to a small change of a variable which they are depending of. Mathematically this can be represented as a Taylor series expansion.[3]

$$
\begin{aligned}
C_L &= f(\alpha, \eta, q, \dot{\alpha}, ...) \\
C_L = C_{L_0} &+ (\tfrac{\delta X}{\delta \alpha}\alpha + \tfrac{\delta^2 X}{\delta \alpha^2}\tfrac{\alpha^2}{2!} + \tfrac{\delta^3 X}{\delta \alpha^3}\tfrac{\alpha^3}{3!} + ...) \\
&+ (\tfrac{\delta X}{\delta \eta}\eta + \tfrac{\delta^2 X}{\delta \eta^2}\tfrac{\eta^2}{2!} + \tfrac{\delta^3 X}{\delta \eta^3}\tfrac{\eta^3}{3!} + ...) \\
&+ (\tfrac{\delta X}{\delta \eta}q + \tfrac{\delta^2 X}{\delta q^2}\tfrac{q^2}{2!} + \tfrac{\delta^3 X}{\delta q^3}\tfrac{q^3}{3!} + ...) \\
&+ (\tfrac{\delta X}{\delta \dot{\alpha}}\dot{\alpha} + \tfrac{\delta^2 X}{\delta \dot{\alpha}^2}\tfrac{\dot{\alpha}^2}{2!} + \tfrac{\delta^3 X}{\delta \dot{\alpha}^3}\tfrac{\dot{\alpha}^3}{3!} + ...) \\
&+ (...)
\end{aligned}
\tag{1.33}
$$

Since the focus is on probation, the absolute sizes of the variables are small. Due to that fact, most of the higher order elements are very small. Which leads to the fact that all higher order elements can be neglected since their influences are small.

$$C_L = C_{L_0} + \frac{\delta X}{\delta \alpha}\alpha + \frac{\delta X}{\delta \eta}\eta + ... \tag{1.34}$$

This is usually displayed in the flowing format:

$$C_L = C_{L_0} + C_{L_\alpha}\alpha + C_{L_\eta}\eta + ... \tag{1.35}$$

1.5.1 Influences of Mach Number

The Mach number is a aerodznamic parameter which is signifiacntly dependent on the fluid and its temperature. The fundamental equations are described in the following :

$$M = \frac{v}{a} \tag{1.36}$$

$$a = \sqrt{\kappa RT} = \sqrt{\kappa \frac{p}{\rho}} \tag{1.37}$$

$$\kappa_{Air} = 1,4$$

$$R_{Air} = 278.058 \tfrac{J}{kgK}$$

Mach Number on Lift

It is commonly known that the aerodynamic moments and forces which are attacking on a missile are hugely depending on the value of the angle of attack which means that with an growing incidence angle, the moments and forces are changing. A similar effect is happening because of a change in the Mach number. With a rising Mach number the lift is also growing. But this has also further changes in the systems behaviour. For the example of the lift performance in Figure 1.5, shows that the Mach number effect reduces the maximum Lift a wing could attain and also decreases the maximum AoA. .[4; 2] The influence of the Mach number on the aerodynamic behaviour can also be seen in the Drag Force and in moments, which makes the aerodynamic coefficients also depend on the Mach number.

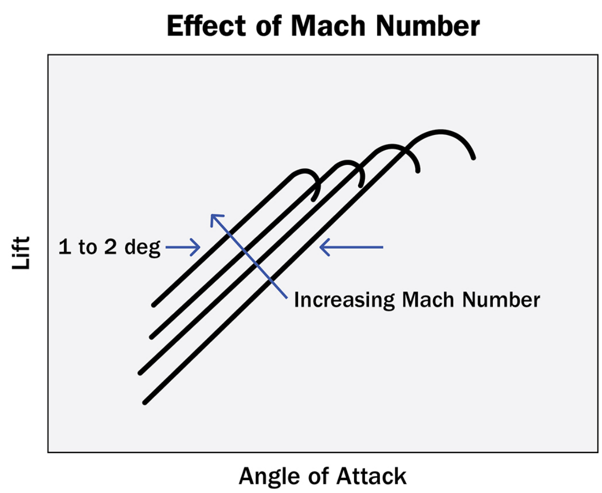

Figure 1.5: Th eLift influences of the AoA and Mach [4]

1.6 Translational Dynamics of given Missile

As priorly derivated and described, can the translational movement of a missile be explained by applying the knowledge of Newtonian dynamics.

$$mD^B v_B^E + m\Omega^{EE} v_B^E = f \tag{1.38}$$

The variableυ_B^E describes the linear velocity state vector of the vehicle while Ω^{EE} is the angular velocity matrix of the vehicle body frame.

$$\left[\upsilon_B^E\right] = \begin{bmatrix} u & v & w \end{bmatrix} \tag{1.39}$$

The external foces are described with f in which describes the aerodynamic and propulsive forces and f^gdescribes the gravitational foces.

$$f = f^a + f^g \tag{1.40}$$

The prior equations projected on the body frame is:

$$\left[\frac{d\upsilon_B^E}{dt}\right]^B = \left[m\Omega^{EE}\right]^B \left[\upsilon_B^E\right]^B + \left(\frac{1}{m}\right)[f^g]^B \tag{1.41}$$

The angular velocity is described as:

$$\left[\omega^{EE}\right] = \begin{bmatrix} p & q & r \end{bmatrix} \tag{1.42}$$

The angular velocity stae vector can be described as:

$$\left[m\Omega^{EE}\right] = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{1.43}$$

The prior defined foces can be summed up and than explained as unified force vector in the body frame.

$$[f]^B = \begin{bmatrix} f_1^B & f_2^B & f_3^B \end{bmatrix} \tag{1.44}$$

By merging all priorly defined equations, the final nonlinear state equations in the body frame are obtained.

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \left(\frac{1}{m}\right) \begin{bmatrix} f_1^B \\ f_2^B \\ f_3^B \end{bmatrix} + \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{1.45}$$

Since the body frame is not suitable for the purpose of the development of a control application since aerodynamics are relevant for that purpose. The wind coordinate system and its state variables should be utilised. The wind cs state variabes are:

- •V is the vehicle Airspeed

$$V = \sqrt{u^2 + v^2 + w^2} \tag{1.46}$$

- •α is the vehicle Angle-of-Attack (AoA)

$$\alpha = atan\left(\frac{w}{u}\right) \tag{1.47}$$

- •β is the vehicle Angle-of-Sideslip (AoS)

$$\beta = atan\left(\frac{v}{\sqrt{u^2 + v^2 + w^2}}\right) \tag{1.48}$$

The translational dynamics variables can be also described as functions of the aerodynamic state variables:

$$u = V\cos\alpha\sin\beta \tag{1.49}$$

$$v = V\sin\beta \tag{1.50}$$

$$w = V\sin\alpha\cos\beta \tag{1.51}$$

Differentiation concerning the time of the equations delivers:

$$\dot{V} = \frac{d}{dt}\{V\} = \frac{u\dot{u} + v\dot{v} + w\dot{w}}{V} \tag{1.52}$$

$$\dot{\alpha} = \frac{d}{dt}\{\alpha\} = \frac{\dot{w}u - \dot{u}w}{u^2 + w^2} \tag{1.53}$$

$$\dot{\beta} = \frac{d}{dt}\{\beta\} = \frac{uv\dot{u} + \left(u^2 + w^2\right)\dot{v} + vw\dot{w}}{V^2\sqrt{u^2 + w^2}} \tag{1.54}$$

The final equations which are describing the time depending behaviour are:

$$\begin{bmatrix} \dot{V} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} 0 \\ q + r(\cos\alpha\tan\theta - \sin)\tan\beta \\ -r(\cos\alpha + \sin\alpha\tan\theta) \end{bmatrix} + \begin{bmatrix} V\cos\alpha\cos\beta & V\sin\beta & V\sin\alpha\cos\beta \\ V\sin\alpha/\cos\beta & 0 & \cos\alpha/\cos\beta \\ -V\cos\alpha\sin\beta & \cos\alpha & -\sin\alpha\sin\beta \end{bmatrix} \frac{\left[f^B\right]^T}{mV} \tag{1.55}$$

The external forces as a vector can be described as the sum of the aero/propulsion and gravity.

$$\begin{bmatrix} f_1^B \\ f_2^B \\ f_3^B \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + mg \begin{bmatrix} -\sin\theta \\ \sin\phi\cos\theta \\ \cos\phi\cos\theta \end{bmatrix} \tag{1.56}$$

The kinematics between the inertial frame I and the body frame B can be described with the Euler angles ϕ,θ,ψ (roll-pitch-yaw) given by:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & \sin\phi \\ 0 & \sin\phi\sec\theta & -\cos\phi\sec\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{1.57}$$

Simplification for longitudinal movement

The given assignment is just focusing on the longitudinal behaviour in its trimming point of the missile system and therefore following assumptions can be made:

- no thrust influences
- $\beta \cong 0$
- neglactable gravitational and axial forces

The relvant aerodynamic forces which are influencing the movement of interest can be described as:

$$Z = \bar{q}SC_N(\alpha, M, \delta_m) \tag{1.58}$$

$$Y = \bar{q}SC_Y(\beta, M, \delta_n) \tag{1.59}$$

With the explained assumptions the state variables time behaviour can be described as:

$$\dot{\alpha} = q + \left(\frac{1}{mV}\right)\cos\alpha Z \tag{1.60}$$

$$\dot{\beta} = -r + \left(\frac{1}{mV}\right)\cos\beta\, Z \tag{1.61}$$

The Taylor Expansion of the aerodynamic perubation behaviour is:

$$C_N\left(\alpha, M, \delta_m\right) = C_N\left(\alpha, M\right)\alpha + C_N\left(M\right)\delta_m \tag{1.62}$$

$$C_Y\left(\beta, M, \delta_n\right) = C_Y\left(\beta, M\right)\beta + C_Y\left(M\right)\delta_n \tag{1.63}$$

The final simplified model is therefore:

$$\dot{\alpha} = q + \left(\frac{\bar{q}S}{mV}\right)\cos\alpha\, C_N\left(\alpha, M, \delta_m\right) \tag{1.64}$$

$$\dot{\beta} = -r + \left(\frac{\bar{q}S}{mV}\right)\cos\beta\, C_Y\left(\beta, M, \delta_n\right) \tag{1.65}$$

1.7 Attitude Dynamics

As for the translational already been demonstrated, also for the rotational movement of the system, the Newtonian dynamics can be utilised.[8]

$$I_B^B D^B \Omega^{BE} + \Omega^{BE} I_B^B \omega^{BE} = m \tag{1.66}$$

The variable $\omega^{BE}$is the angular velocity state vector of the vehicle body frame, and Ω^{BE} is the angular velocity of the vehicle body frame. The attacking moment is described m and I is the vehicle moment of inertia.The external moment components can elementwise be described as:

$$[m]^B = \begin{bmatrix} m_1^B & m_2^B & m_3^B \end{bmatrix} \tag{1.67}$$

The moment of inertia tensor of the missile body as priorly explained:

$$\left[I_B^B\right]^B = \begin{pmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{pmatrix} \tag{1.68}$$

The prior equations projected on the body frame is:

$$\left[\frac{d\omega_B^E}{dt}\right]^B = \left(\left[I_B^B\right]^B\right)^{-1} \{-\left[\Omega^{EE}\right]^B \left[I_B^B\right]^B \left[\omega^{BE}\right]^B + [m]^B\} \tag{1.69}$$

The final equations which are describing the time depending behaviour are:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \left(\frac{1}{m}\right) \begin{bmatrix} \left(\frac{I_2-I_3}{I_1}\right) qr \\ \left(\frac{I_3-I_1}{I_2}\right) pr \\ \left(\frac{I_1-I_2}{I_3}\right) pq \end{bmatrix} + \begin{bmatrix} \left(\frac{1}{I_1}\right) & -r & 0 \\ 0 & \left(\frac{1}{I_2}\right) & 0 \\ 0 & 0 & \left(\frac{1}{I_3}\right) \end{bmatrix} \begin{bmatrix} m_1^B \\ m_2^B \\ m_3^B \end{bmatrix} \tag{1.70}$$

The relvant aerodynamic moments which are influencing the movement of interest can be described as:

$$L = \bar{q} S d C_m (\alpha, M, \delta_m) \tag{1.71}$$

$$N = \bar{q} S d C_n (\beta, M, \delta_n) \tag{1.72}$$

Also for the rotational dynamics a set of assumptions can be made in order to simplify the complex problem:

- $p \cong 0$

The simplified dynamcs than can be described as:

$$\dot{q} = \left(\frac{1}{I_3}\right) M \tag{1.73}$$

$$\dot{r} = -\left(\frac{1}{I_3}\right) N \tag{1.74}$$

With a apllication of the taylor expansion on the aerodynamics pertination moment we receive:

$$C_m(\alpha, M, \delta_m) = C_{m\alpha}(\alpha, M)\,\alpha + C_{m\delta}(M)\,\delta_m \tag{1.75}$$

$$C_n(\beta, M, \delta_n) = C_{n\beta}\beta, M\beta + C_{n\delta}(M)\,\delta_n \tag{1.76}$$

The final simplified model:

$$\dot{q} = \left(\frac{\bar{q}Sd}{I_3}\right) C_{m\alpha}(\alpha, M, \delta_m) \tag{1.77}$$

$$\dot{r} = \left(\frac{\bar{q}Sd}{I_3}\right)\left(\frac{\bar{q}S}{mV}\right) C_n(\beta, M, \delta_n) \tag{1.78}$$

1.7.1 Mach Dynamics

The Mach number time behaviour, which is considered to be slowly changing, and can be explained with:

$$\begin{gathered}\dot{M} = \left(\frac{1}{a}\right)\left[-g|n_z|\sin|\alpha| + A_x M a^2 \cos\alpha\right] \\ = \left(\frac{1}{a}\right)\left[A_x M a^2 \cos\alpha + g n_z \cos\alpha\right] \\ = \left(\frac{1}{a}\right)\left[\left(\frac{1}{m}\right)(0.7 P_0 M a^2) S C_x \cos\alpha + \left(\frac{g}{mq}\right) Z \sin\alpha\right] \\ = \left(\frac{1}{ma}\right)\left[X\cos\alpha + Z\sin\alpha\right]\end{gathered} \tag{1.79}$$

Since we know that $M = \frac{V}{a}$ and with the assumption that $\beta \cong 0$ we finally get the axial dynamics:

$$\dot{V} = \left(\frac{1}{m}\right)[(X\cos\alpha + Z\sin\alpha] \tag{1.80}$$

1.7.2 System Represenation

In a generic form, the longitudinal system movement (pitch axis state & output) can be described as a Nonlinear Parameter Dependent System (NLPD) in the following format:

$$\dot{x} = f_x\left[x(t), u(t), \sigma(t)\right] \tag{1.81}$$

$$y = f_y\left[x(t), u(t), \sigma(t)\right] \tag{1.82}$$

$$\dot{\sigma} = f_\sigma\left[x(t), u(t), \sigma(t)\right] \tag{1.83}$$

An equivalent way to describe the system is,

$$x = \begin{bmatrix} \alpha \\ q \end{bmatrix}, u = \delta_m, x = \begin{bmatrix} n_z \\ q \end{bmatrix}, \sigma = \mathscr{M} \tag{1.84}$$

1.8 Equilibrium analysis

1.8.1 Hartman- Grobman Theorem

To analyse the dynamic characteristic of a system is needed to reduce the complexity of complex and highly non-linear systems. The Hartman- Grobman Theorem presents a tool which allows it to understand a systems dynamic behaviour with a simplified approach. It says that a response of a system in a region close to a hyperbolic equilibrium spot is qualitatively the identical as the behaviour of its linearization near this equilibrium location. A hyperbolic equilibrium is defined as a point in which the eigenvalues of the linearization have a real part that is zero. This means that this theorem allows us to linearise a model to analyse its behaviour without losing relevant information.[5; 1; 2]

1.8.2 Equilibirum Analysis

In order to find an equilibrium point the following conditions need to be found for a given equation system.

$$\dot{x} = f_x\left[x(t), u(t), \sigma(t)\right] \tag{1.85}$$

$$\dot{x} = 0 \tag{1.86}$$

$$\bar{y} = f_y\left[x(t), u(t), \sigma(t)\right] \tag{1.87}$$

.

1.9 Equilibirum Analysis of Missile application

Since we have, in our case, more unknown variables as equations it is needed to impose two variables to match the number of unknown variables and equations. In general, the Mach number is a variable which is commonly imposed, since it provides a slow dynamic change over time and can be assumed as static. The second variable, which needs to be imposed in order to get an equilibrium state, can be chosen depending on the values of interest.

$$\begin{gathered} q+\left(\frac{\bar{q}(\mathscr{M})S}{mV}\right)\cos\alpha\, C_n\left(\alpha, M, \delta_m\right)=0 \\ \cancel{\left[\frac{\bar{q}(\mathscr{M})Sd}{I2}\right]} C_m\left(\alpha, M, \delta_m\right)=0 \\ n_z-\left[\frac{\bar{q}(\mathscr{M})S}{mg}\right] C_N\left(\alpha, M, \delta_m\right)=0 \end{gathered} \tag{1.88}$$

The state and equilibrium point can be defined as:

$$\rho_x=\left[\begin{array}{c}\alpha \\ q\end{array}\right] or\, \rho_x=\left[\begin{array}{c}n_z \\ q\end{array}\right] \tag{1.89}$$

The computing of the equilibrium point can be decomposed in three distrinct steps.

- Step 1: Solve for the control from the output equation as a function of the equilibrium point vector (known) and AoA (unknown)
- $$\delta_m=\delta_m\left(\rho_y, \alpha\right) \tag{1.90}$$
- Step 2: Replace the control inside the moment equation and solve for the AoA and for the control
- $$\begin{gathered} C_m\left[\alpha, \mathscr{M}, \delta_m(\rho_y, \alpha)\right]=0 \\ C_m\left[\rho_y, \alpha\right]=0 \\ \alpha=f_\alpha(\rho_y), \delta_m=\delta_m\left[(\rho_y, f_\alpha(\rho_y)\right] \end{gathered} \tag{1.91}$$
- Step 3: Replace the AoA and the control inside the force equation to compute the pitch rate at equilibrium
- $$q=f_q(\rho_y) \tag{1.92}$$

1.9.1 Jacobian Linearization

The Hartman- Grobman Theorem showed us that it is possible to simplify a model with a linearization without losing relevant information about the dynamic behaviour at a particular operating point. In the following, the basic linearisation approach is explained.

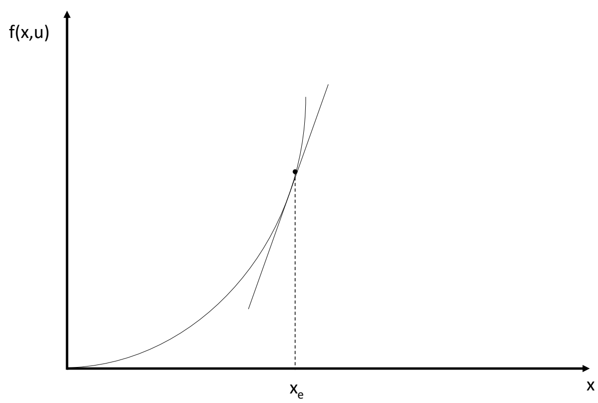

Figure 1.6: A conceptual sketch of the linearisation appraoch

The non-linear equation can be described as:

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

The equalibrium would be:

$$x = x_0$$

Small pertubation around that operations point are:

$$x = x_0 + \delta x$$

The Taylor-Series Approximation around the pertubation is described followed :

$$\dot{x} + \delta\dot{x} = f(x_0 + \delta x) = f(x_0) + \frac{\partial f}{\partial x} + \frac{\partial^2 f}{\partial x^2}\frac{(\partial x)^2}{2!} + \dots$$

Since we dediced to that higher orders of the equation can be neglected.The Jacobean linearization deliveres following equations:

$$\begin{array}{lll} A(\rho_y) = \frac{\partial f_x}{\partial x}|_{eq} & = \frac{\partial f_x}{\partial x}|_{\bar{x}(\rho_y),\bar{u}(\rho_y)} & = \frac{\partial f_x}{\partial x}|_{\rho_y} \\ B(\rho_y) = \frac{\partial f_x}{\partial x}|_{\rho_y} & C(\rho_y) = \frac{\partial f_x}{\partial x}|_{\rho_y} & D(\rho_y) = \frac{\partial f_x}{\partial x}|_{\rho_y} \end{array}$$

1.9.2 Linearization of given System

The relevant state equations for the given problem are:

$$\dot{\alpha} = f_\alpha = q + \left(\frac{\bar{q}S}{mV}\right) \cos\alpha\, C_N\left(\alpha, M, \delta_m\right) + q \tag{1.93}$$

$$\dot{\beta} = f_q = \left(\frac{\bar{q}Sd}{mV}\right) C_m\left(\alpha, M, \delta_n\right) \tag{1.94}$$

$$n_z = f_{n_z} = \left(\frac{\bar{q}S}{mg}\right) C_N\left(\alpha, M, \delta_m\right) \tag{1.95}$$

An application of the linearisation aproach on the given state equation is giving following results:

$$\frac{\partial f_\alpha}{\partial \alpha} = \left(\frac{\bar{q}S}{mV}\right) \left[\frac{\partial C_N}{\partial \alpha} \cos\alpha - C_N \sin\alpha\right] = \frac{Z_\alpha}{V} \tag{1.96}$$

$$\frac{\partial f_{n_z}}{\partial \delta_m} = \left(\frac{\bar{q}S}{mV}\right) C_N \cos\alpha = \frac{Z_\delta}{V} \tag{1.97}$$

$$\frac{\partial f_q}{\partial \alpha} = \left(\frac{\bar{q}Sd}{I_2}\right) \frac{\partial C_m}{\partial \alpha} = M_\alpha \tag{1.98}$$

$$\frac{\partial f_q}{\partial q} = \left(\frac{\bar{q}Sd}{I_2}\right) \left(\frac{d}{V}\right) C_{mq}\left(\mathscr{M}\right) = M_q \cong 0 \tag{1.99}$$

$$\frac{\partial f_{n_z}}{\partial \alpha} = \left(\frac{\bar{q}S}{mg}\right) \frac{\partial C_N}{\partial \alpha} = \frac{A_\alpha}{g} \tag{1.100}$$

$$\frac{\partial f_{n_z}}{\partial \delta_m} = \left(\frac{\bar{q}S}{mg}\right) \frac{\partial C_N}{\partial \delta} = \frac{A_\delta}{g} \tag{1.101}$$

1.10 State Space Linearized Model

Thederivated final state space linearized model has the following standard form:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \end{bmatrix} = \left(\frac{1}{m}\right) \begin{bmatrix} \frac{Z_\alpha}{V} & 1 \\ M_\alpha & M_q \end{bmatrix} \begin{bmatrix} \alpha \\ \delta \end{bmatrix} + \begin{bmatrix} \frac{Z_\alpha}{V} \\ M_\delta \end{bmatrix} \delta_{m,\delta} \tag{1.102}$$

Chapter 2

Uncertain System Modelling

2.1 Introduction

In order to investigate the missile system and to develop a suitable control system for it, we need to create an appropriate dynamic system which can reflect the system dynamics. Since we have to deal with uncertainties in our given problem, the system also has to respect defined uncertainties in its approach. This section will show how both, a ideal system and one with uncertainties, has been implemented in the Matlab environment and how they both differ in their behaviour.

2.2 Approach in MATLAB/Simulink

As a first step, the systems (nominal and uncertain) had to be implemented. From the given information we received the state space representation for the system displayed in figure2.1.

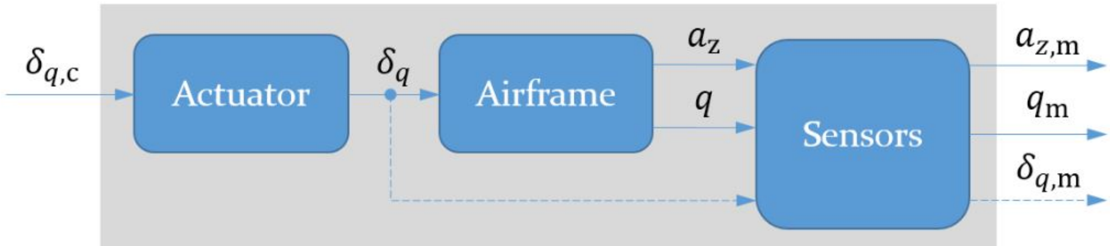

Figure 2.1: Sketch of the Open Loop Diagram

In order to model a valid state space system relevant information matrices of A (System matrix), B (Input matrix), C (Output matrix) and D (Feedforward matrix) had to defined and merged into a state-space system by using the 'SS()' Matlab function. For the airframe, the prior derivated state space representation has been used. The actuator has been defined in the following way:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_a^2 & -2\zeta_a\omega_a \end{bmatrix} \begin{bmatrix} \delta_q \\ \dot{\delta}_q \end{bmatrix} + \begin{bmatrix} 0 \\ \omega_a^2 \end{bmatrix} \delta_{q,c} \tag{2.1}$$

The Sensor has been defined as ideal observers, which leads to the following state space representation:

$$\begin{bmatrix} a_{z_m} \\ q_m \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} a_{z_m} \\ q_m \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a_z \\ q_m \end{bmatrix} \tag{2.2}$$

The airframe is the system that has to deal with the following uncertainties:

$$M_\alpha(1 - r_{M_\alpha}) \leq M_\alpha \leq (1 + r_{M_\alpha})M_\alpha \tag{2.3}$$

$$M_\delta(1 - r_{M_\delta}) \leq M_\delta \leq (1 + r_{M_\delta})M_\delta \tag{2.4}$$

The uncertainties can be defined by using the 'UREAL()' function of the robust control toolbox, which allows to define and respect uncertainties during the computing of an uncertain system.

2.3 Comparision of Uncertain and Nominal System Behaviour

In order to receive a further understanding of the differences between the different dynamic characteristics of an uncertain and idealised system, both systems had been compared in different methodologies.

2.3.1 Zero and Poles of the Systems

In order to investigate the stability behaviour and possible non-minimal-phase zeros, for both systems, the zero and poles have been plotted as z-plane diagrams.

Output: a_z

Figure 2.2 shows the nominal and uncertain zero and poles locations in the z-plane for the open loop system nad its outout of a_z. It can be seen that the locations of the zeros and poles for the nominal system are on a fixed place, while the zeros and poles of the uncertain system are distributed in a certain area.The system dynamic behaviour is in both cases stable, but the system has a non minimum phase zero, which has it's orgin from the airframe system.

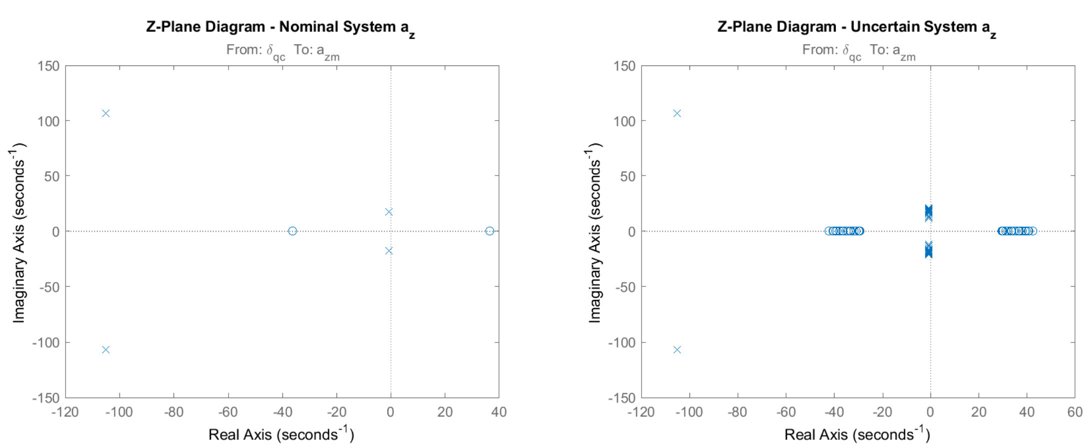

Figure 2.2: Z-Plane Plot of the nominal (left) and uncertain System Dynamics for the Output a_z

Output: q

Figure 2.3 shows the nominal and uncertain zero and poles locations in the z-plane for the open loop system nad its outout of q. It can be seen that the locations of the zeros and poles for the nominal system are on a fixed place, while the zeros and poles of the uncertain system are distributed in a certain area.The system dynamic behaviour is in both cases stable and the system has a no non-minimum phase zersos for this output.

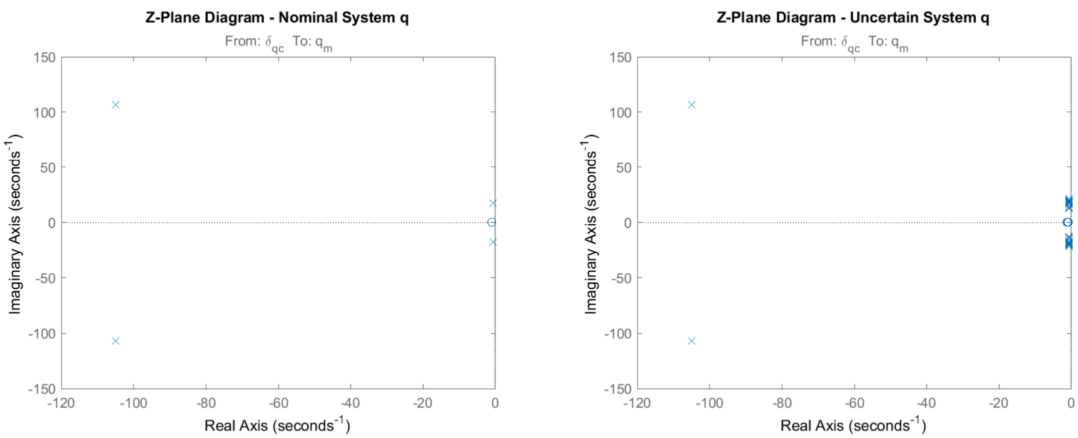

Figure 2.3: Z-Plane Plot of the nominal (left) and uncertain System Dynamics for the Output q

2.3.2 Comparision of Bode Diagrams

The Bode plot of both nominal and uncertain outputs (a and q) are given in figure 2.4. Both output values are showing a peak at a certain frequency in their Magnitude function. This indicates that the open loop system can be understood be as underdamped. An underdamped answer is one that oscillates within a crumbling envelope. The more underdamped the system, the larger the oscillation gains and the more it needs to achieve steady-state.**(author?)** [7]

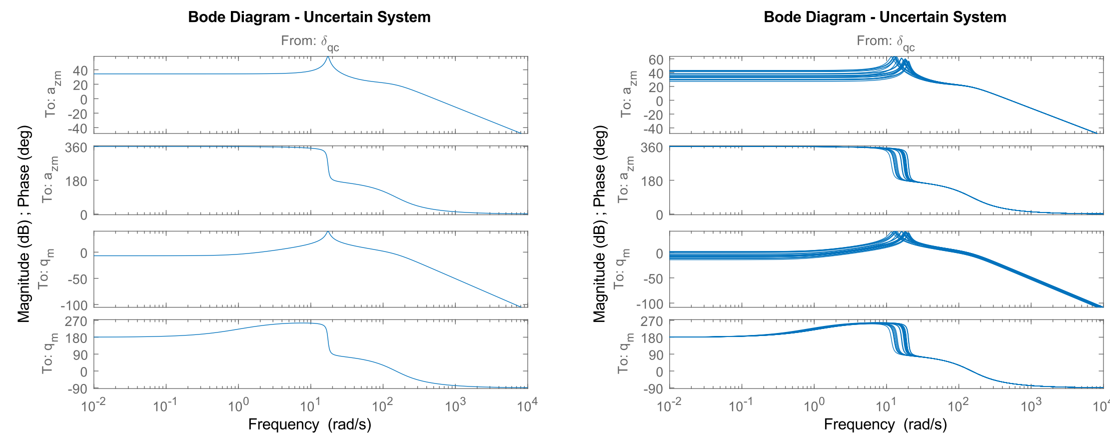

Figure 2.4: Bode Plot of the nominal (left) and uncertain System Dynamics for the Output a_z and q

2.3.3 Comparision of Impulse Responses

The impulse responses in figure 2.5 shows that the system gives out a bounded output for both parameters at the beginning. From there it oscillates around the mean of zero until it decays to a nullity output.

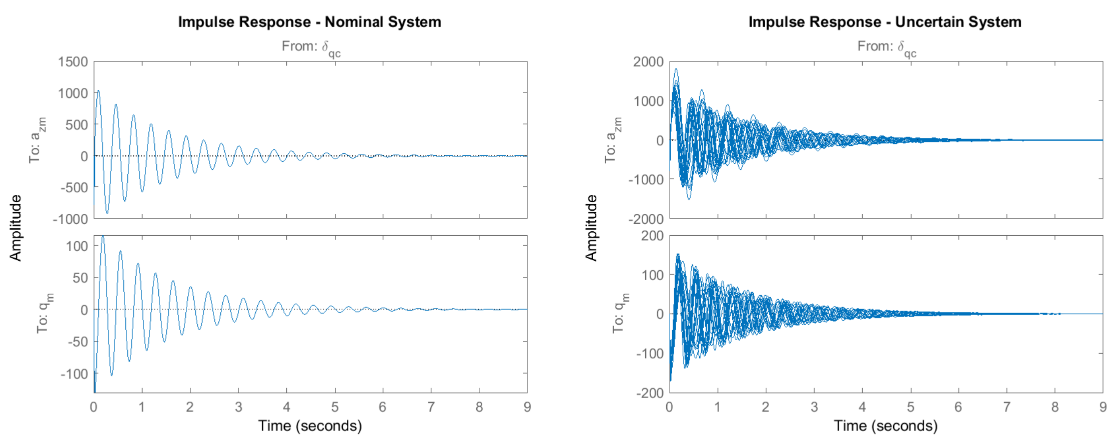

Figure 2.5: Impulse Response Plot of the nominal (left) and uncertain System Dynamics for the Output $a_z$and q

2.3.4 Comparision of Step Responses

The step responses in figure 2.6 shows that the system has a similar output for both parameters like the impulse response. After reaching an absolute max. Output value. The system oscillates around the mean of a non-zero mean until it decays to a nullity output.

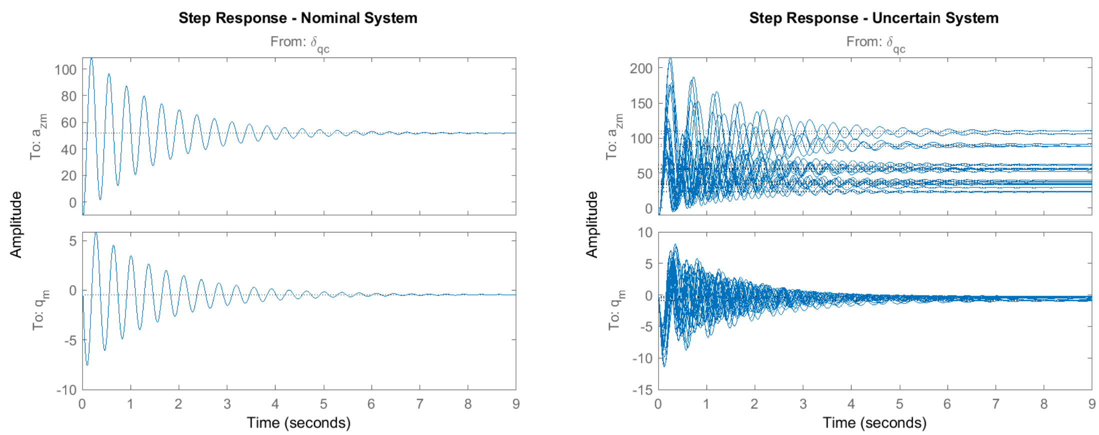

Figure 2.6: Step Response Plot of the nominal (left) and uncertain System Dynamics for the Output $a_z$and q

2.4 Linear Fraction Transformation

2.4.1 Approach in MATLAB/Simulink

In order to compute the LFT matrix the 'LFTDATA' of the robust control toolbox has been utilised. The function returns a fixed specific part and a normalised uncertain part.

2.4.2 Expected Results

With regards to the lecture in robust control, follwing results were expected from the use of 'LFTDATA()':

$$H(s) = \begin{bmatrix} \begin{pmatrix} \dot{\alpha} \\ \dot{\alpha} \end{pmatrix} \\ \begin{pmatrix} y_{M_\alpha} \\ y_{M_\delta} \end{pmatrix} \\ \begin{pmatrix} n_z \\ q \end{pmatrix} \end{bmatrix} = \begin{bmatrix} \begin{pmatrix} Z_\alpha & 1 \\ \bar{M}_\alpha & \overline{M}_q \end{pmatrix} & \begin{pmatrix} 0 & 0 & Z_\delta \\ \sqrt{\overline{M_\alpha} r_{M_\alpha}} & \sqrt{\overline{M_\delta} r_{M_\delta}} & M_\delta \end{pmatrix} \\ \begin{pmatrix} \sqrt{\overline{M_\alpha} r_{M_\alpha}} & 0 \\ 0 & 0 \\ A_\alpha & 0 \\ 0 & 01 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & \sqrt{\overline{M_\delta} r_{M_\delta}} \\ 0 & 0 & A_\delta \\ 0 & 0 & 0 \end{pmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} \begin{pmatrix} -1231.914 & 1 \\ -299.26 & -130.866 \end{pmatrix} & \begin{pmatrix} 0 & 0 & -1231.914 \\ 13.15 & 6.543 & -130.9 \end{pmatrix} \\ \begin{pmatrix} 13.15 & 0 \\ 0 & 0 \\ -1429.131 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 6.543 \\ 0 & 0 & -114.59 \\ 0 & 0 & 0 \end{pmatrix} \end{bmatrix}$$

2.4.3 Computation Results

The utilised function returned following results:

$$H(s) = \begin{bmatrix} \begin{pmatrix} \dot{\alpha} \\ \dot{\alpha} \end{pmatrix} \\ \begin{pmatrix} y_{M_\alpha} \\ y_{M_\delta} \end{pmatrix} \\ \begin{pmatrix} n_z \\ q \end{pmatrix} \end{bmatrix} = \begin{bmatrix} \begin{pmatrix} Z_\alpha & 1 \\ \bar{M}_\alpha & \overline{M}_q \end{pmatrix} & \begin{pmatrix} 0 & 0 & Z_\delta \\ \sqrt{\overline{M_\alpha}} r_{M_\alpha} & \sqrt{\overline{M_\delta}} r_{M_\delta} & M_\delta \end{pmatrix} \\ \begin{pmatrix} \sqrt{\overline{M_\alpha}} r_{M_\alpha} & 0 \\ 0 & 0 \\ A_\alpha & 0 \\ 0 & 01 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & \sqrt{\overline{M_\delta}} r_{M_\delta} \\ 0 & 0 & A_\delta \\ 0 & 0 & 0 \end{pmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} \begin{pmatrix} -1.3 & 1 \\ -299.26 & -130.866 \end{pmatrix} & \begin{pmatrix} 0 & 0 & -0.1136 \\ 13.15 & 6.543 & -130.9 \end{pmatrix} \\ \begin{pmatrix} 13.15 & 0 \\ 0 & 0 \\ -145.7 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 6.543 \\ 0 & 0 & -11.68 \\ 0 & 0 & 0 \end{pmatrix} \end{bmatrix}$$

2.4.4 Comparision

It was observed that the expected values are not equal to the computed once. After a analysis of the differences it was assumed that possible miss spellings in the slides could have lead to that difference. Therefore was the expected matrix adjusted in the follwing way:

$$H(s) = \begin{bmatrix} \begin{pmatrix} \dot{\alpha} \\ \dot{\alpha} \end{pmatrix} \\ \begin{pmatrix} y_{M_\alpha} \\ y_{M_\delta} \end{pmatrix} \\ \begin{pmatrix} n_z \\ q \end{pmatrix} \end{bmatrix} = \begin{bmatrix} \begin{pmatrix} \frac{Z_\alpha}{V} & 1 \\ \bar{M_\alpha} & \overline{M}_q \end{pmatrix} & \begin{pmatrix} 0 & 0 & \frac{Z_\delta}{V} \\ \sqrt{\overline{M_\alpha}} r_{M_\alpha} & \sqrt{\overline{M_\delta}} r_{M_\delta} & M_\delta \end{pmatrix} \\ \begin{pmatrix} \sqrt{\overline{M_\alpha}} r_{M_\alpha} & 0 \\ 0 & 0 \\ \frac{A_\alpha}{g} & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & \sqrt{\overline{M_\delta}} r_{M_\delta} \\ 0 & 0 & \frac{A_\delta}{g} \\ 0 & 0 & 0 \end{pmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} \begin{pmatrix} -1231.914 & 1 \\ -299.26 & -130.866 \end{pmatrix} & \begin{pmatrix} 0 & 0 & -1231.914 \\ 13.15 & 6.543 & -130.9 \end{pmatrix} \\ \begin{pmatrix} 13.15 & 0 \\ 0 & 0 \\ -1429.131 & 0 \\ 0 & 1 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 6.543 \\ 0 & 0 & -114.59 \\ 0 & 0 & 0 \end{pmatrix} \end{bmatrix}$$

After the adjustments both, the comuted and expected, were equal.

Chapter 3

Outter Loop Unstructured Uncertainty

3.1 Approach in MATLAB/Simulink

In order to create the outer feedback loop of the uncertain plant dynamics, the 'SS()' and 'CONNECT()' functions of the control toolbox has been utilised. The functions allow it to connect the prior defined state space blocks properly and to allow an analysis of the assembly.

3.1.1 Stabilty Analysis

In order to create an appropriate feedback system, we need to analiys the systems stabilty in way which allows to identify which gains could lead to an undesired system behaviour.

Nyquist Stabilty Analysis

To find a suitable gain of Kq the first step was to analyse the Nyquist diagram (shown in figure 3.1). The blue plot shows the open loop with a positive gain and orange one shows the system with a negative gain. It can be seen that blue system is not stable since it turns around the critical point clockwise. So it is needed to close the loop with a negative gain, this gain will stabilise the given system. This approach can be interpreted as an application of the Nyquist criterion. The Nyquist criterion delivers for our case, in which we have 0 unstable poles and 1 non-minimum phase zero, that the system must turn anticlockwise around the critical point (0-1 = -1). This is only the case with a negative gain. With a positive gain, your system is turning -2 times around the critical point. This would lead to a closed loop with unstable behaviour. round the critical point.

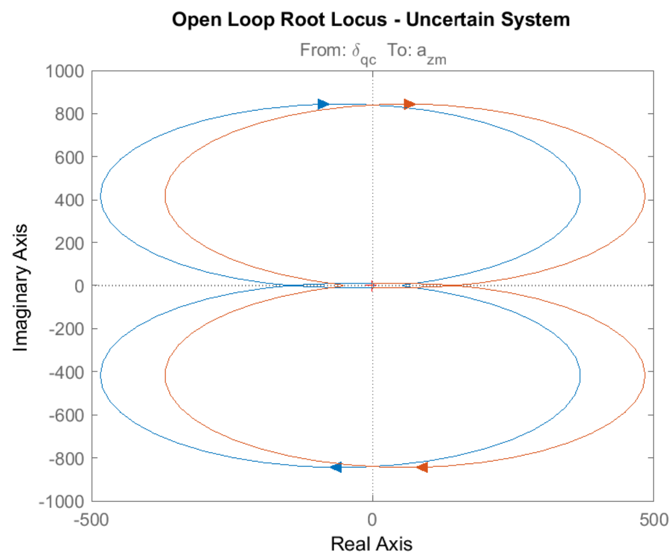

Figure 3.1: Nyquist Plot of Open Loop System

Step Response Analysis

The prior given theoretical explanation can be also validated with an investigation of the step responses of the closed-loop dynamics with a positive and negative unitary gain feedback in figure 3.2. The diagrams show that the system behaviour for a positive feedback gain is unstable (left), while it is stable (right) for the case with a negative gain.

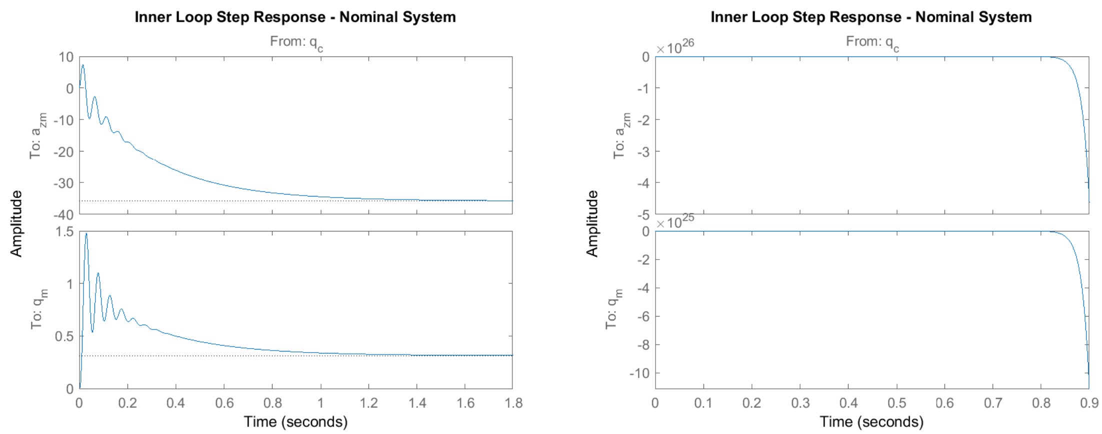

Figure 3.2: Step resonse of closed inner loop with negative gain (left) and positive gain (right)

3.1.2 Root Locus Diagram

One of the given design requirements is it to create a feedback system in which the dominant pole has a damping ratio of 0.707. This has been achieved by analysing the root locus of the system by plotting the target damping line in the graph. This allowed analysing which gain is needed to achieve this behaviour. The gain which has been identified is -0.165.

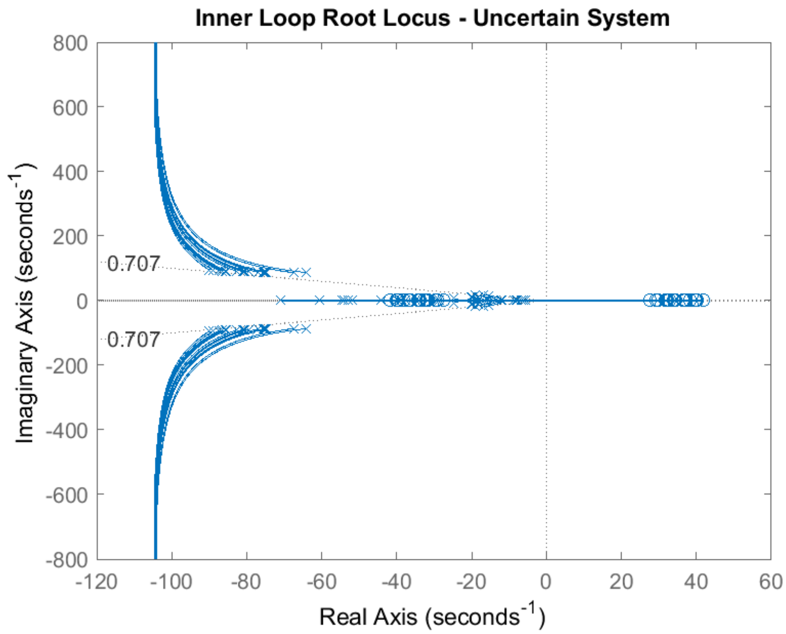

Figure 3.3: A thrust vector attacking on a misiles

3.1.3 Outter Loop Response

Since it is required that the system achieves a unitary steady-state output. A feedforward gain has to be implemented into the system. To achieve that, the controller should invert the plant gain and multiplies it by the target system value to obtain, which is in our case 1.

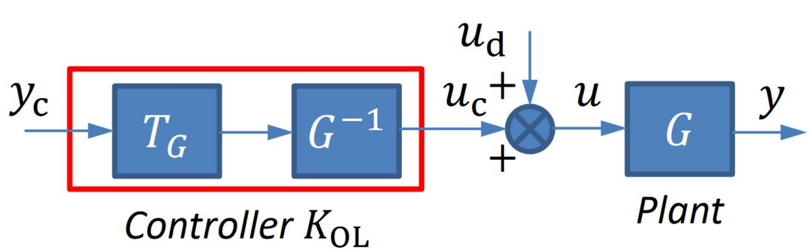

Figure 3.4: A thrust vector attacking on a misiles

To accomplish that behaviour the steady-state gain of the inner loop has been computed by the MATLAB function 'DCGAIN()' and used for the gain of the controller K_{sc} in the following way. This enabled the system to give out a unitary steady-state output.

3.2 Final System Results

In the following, the final system behaviour had to be analysed to proof if the system act like it suppose to be.

3.2.1 Step Respomse of System

The step responses in figure 3.5 shows that the system shows the desired behaviour. That means that the system achieves the steady state value a_z of 1 over a time period of up to 2 seconds. The uncertain model plot on the right side shows how much the system value can differ from time to time, even though the controller gains were designed correctly. The output of a shows that, despite the fact that 1 is the goal, the output could settle between 2 and 0.5.

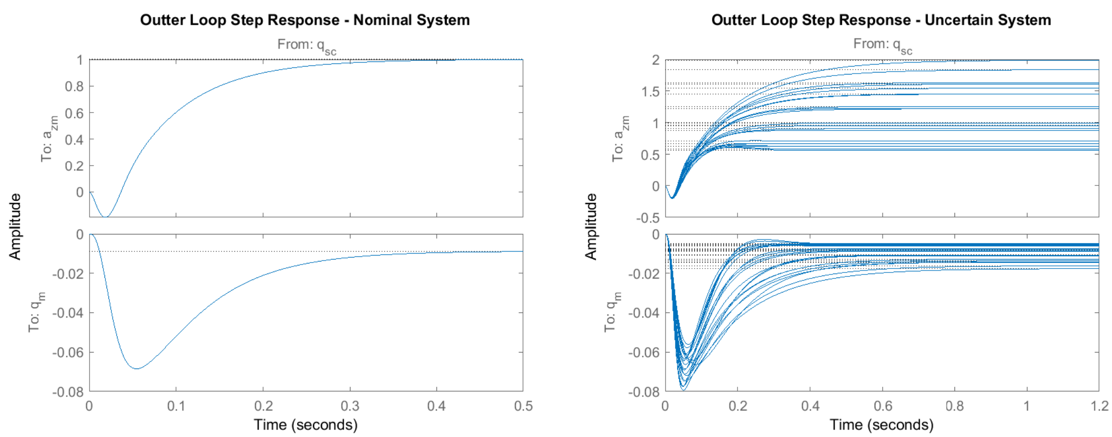

Figure 3.5: Step respinse of outter loop feedback system (nominal and uncertain)

Chapter 4

Uncertain Weight Computation

4.1 Approach in MATLAB/Simulink

Each uncertain object (umat,uss,ufrd) may be represented as a linear fractionaltransformation of non-uncertain part and a matrix containing the uncertain parame-ters. Using the command 'LFTDATA' an uncertain object may be decomposed into anominal part and a normalized uncertainty matrix with H∞norm, equal to 1. For agiven uncertain objectusys-the command line

4.1.1 Uncertainty weight computation - Additive

The goal, from the control design point of view, would it be to find an lm with the smallest possible radius. That means that coverage of the samples is close to the real systems dynamics. In figure 4.1 we can see the plottings for the additive form. It can be seen that big gaps are given and that the different lm functions (order and sample variation) are not clinging to the samples.

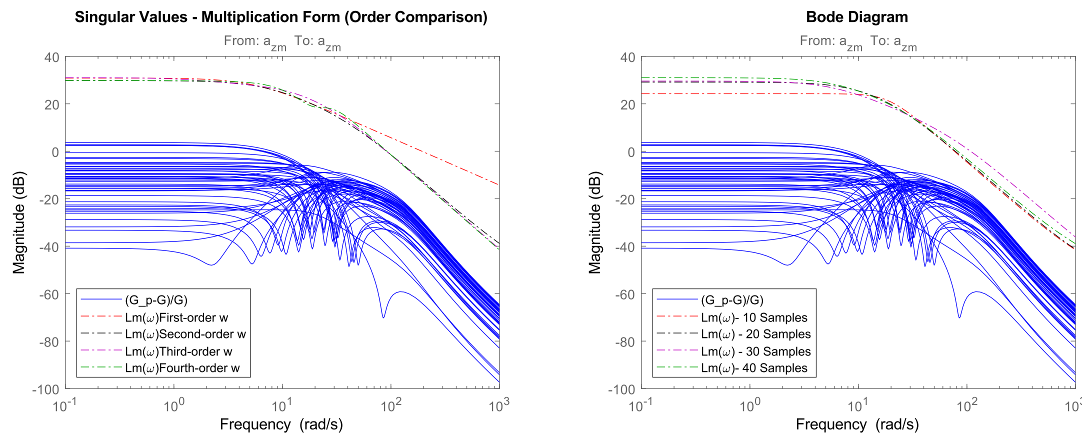

Figure 4.1: Ploted singular Values with Additive Form and variations of Order and Number of Samples

4.1.2 Uncertainty weight computation - Multiplicative

The multiplicative in figure 4.2 is showing better results than the additive approach. The radius is small and close to the dynamics of the system. The order and samples variations are confirming that the higher the order and the number of samples are, the better the results. But a high number of samples and the order of the computing approach are related to a certain computational complexity a trade-off has to be made. Therefore the second-order are chosen for further design since it provides a right balance between the complexity and the accuracy.

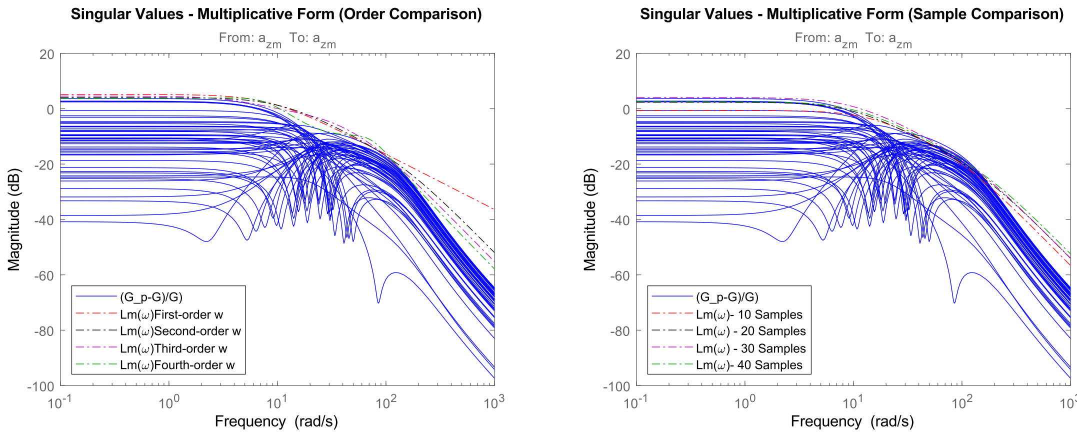

Figure 4.2: Ploted singular Values with Multiplicative Form and variations of Order and Number of Samples

Chapter 5

Control Design

5.1 Verfication of Designed Closed Loop System

The initial step is it to verify the designed inner closed loop system in order to ensure that a working system is used for further steps.

5.1.1 Approach in MATLAB

In order to vefiy the inner closed feedback loop transfer function, the 'ZERO()',' POLES()' and ' TF()' the functions of the standard MATLAB library has been utilised.

5.1.2 Verifaction Results

Obtained Transfer Function

The obtained transfer function from MATLAB is:

$$T_{RM}(s) = \frac{\omega_{RM}(-\frac{s}{Z_{NMP}}+1)}{s^2+2\zeta_{RM}\omega_{RM}s+\omega_{RM}^2} \tag{5.1}$$

The order of the denominator is 4, which gives information about the number of poles of the system. The order of the nominator is 2 and therefore it shows that the system has as required 2 zeros.

Zero and Poles of the System

Eventhough is could be derivated directly from the transfer function that the system has the required number of zeros and poles.Further Z-Plane diagrams were created with the MATLAB plots to deliver a convinient graphical display.

Output: a_z

Figure 5.1 shows the nominal and uncertain zero and poles locations in the z-plane for the open loop system nad its outout of a_z. It can be seen that the locations of the zeros and poles for the nominal system are on a fixed place, while the zeros and poles of the uncertain system are distributed in a certain area.The system dynamic behaviour is in both cases stable, but the system has a non minimum phase zero, which has it's orgin from the airframe system.

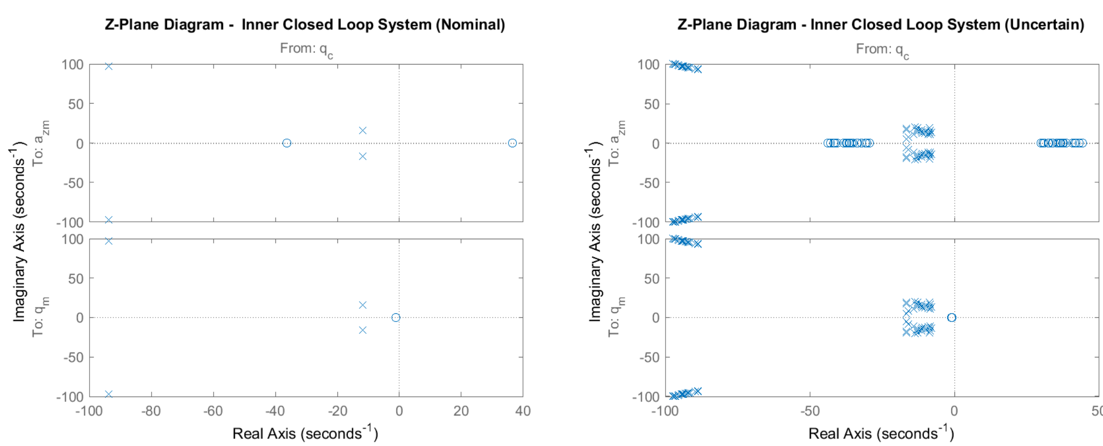

Figure 5.1: Z-Plane Plot of the nominal (left) and uncertain System Dynamics for the Output a_z

The impulse responses in figure 5.2 shows that the system gives out a bounded output for both parameters at the beginning. From there it oscillates around the mean of zero until it decays to a nullity output.

5.1.3 Response of the System

	Results
t_s	$0.2973s$
M_p	12.8950%

Tabelle 5.1: Computed Response Results

The inspected system has been analysed for its dynamic performance in order to allow verification of the behaviour of the system. The dynamic characteristic, such as settling time and overshoot in %, have been extracted from the system by using the MATLAB library function 'stepInformation()'. The results of that are given in table 5.1.

Furthermore, a step function has been applied and the response plotted in figure 5.2.

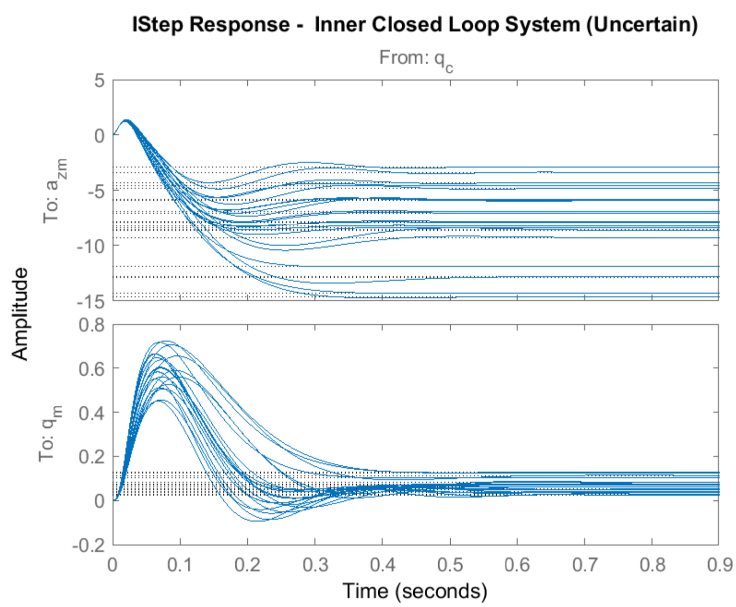

Figure 5.2: Step Response Plot of the uncertain System Dynamics for the Output $a_z$and q_m

Interpretation of Step Response

Figure 5.2 shows that the control system is able to follow the reference parameter, after a given settling time. In the first phase is the system undershooting with a negative value until it is able to follow the lead in the correct way. The undershoot can be explained analytically and in a physical way which is more intuitive to understand. The mathematical explanation of the undershoot is the non-minimum phase zero of the transfer function. This NMP zero has ban inherited from the airframe system which was given. Another explanation can be given through the flight dynamic behaviour of the airframe in order to manoeuvre in the desired way. To be able to pitch down a fin deflection has to be initiated. This creates a small aerodynamic force on the fin, which on the other hand creates large fin aerodynamic moment. These moments are creating a pitch rate which leads to an increase in AoA. This interaction of forces and moments are creating normal acceleration. The undershoot which can be seen in figure 5.2 can be explained with the initial aerodynamic force of the fin which is needed to initiate the required manoeuvre.

Feedforwad Control

The outer loop controller K_{cf} can be seen as a feedforward controller which allows the system to track capability. To achieve this is the gain influencing the control signal to reach the desired output magnitude. The gain in its design idea has no influences on the disturbance rejection. This means we indeed have a functional system that can follow a reference parameter characteristic. But to achieve the desired magnitude a feedforward control gain needs to be utilised.

5.2 Reference Model Computation

The obtained design goal of the system is it to behave like the following second-order transfer function:

$$T_{RM}(s) = \frac{\omega_{RM}(-\frac{s}{Z_{NMP}}+1)}{s^2+2\zeta_{RM}\omega_{RM}s+\omega_{RM}^2} \tag{5.2}$$

The obtained system should be defined in a way in which the 2% settling time should be 0.2 seconds and the maximum overshoot max. 2%.

$$\text{Settling Time: } t_s = 0.2s$$

$$\text{max. Overshoot: } M_p = 2\%$$

5.2.1 Analytical Approach

The required characteristic has been obtained by the use of analytical approaches from the fundamental classical control engineering theory.(MIT)

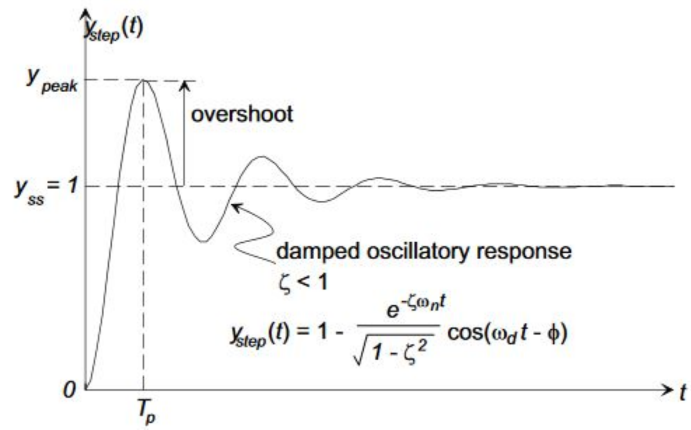

Figure 5.3: A thrust vector attacking on a misiles

The overshoot can be defined in the following way:

$$\%OS = \frac{y_{Peak} - y_{ss}}{y_{ss}} x100 \tag{5.3}$$

The peak at T_p can be described with:

$$y_{Peak} = y(T_p) = e^{-(\zeta\pi/\sqrt{1-\zeta^2})} x100 \tag{5.4}$$

The damping ratio of the dynamic system is :

$$\zeta = \frac{-ln(\%OS/100)}{\sqrt{\pi^2 + ln^2(\%OS/100)}} \tag{5.5}$$

Consequently can be the natural system frequency be computed with:

$$\omega_{RM} = \frac{ln((\%OS/100)\sqrt{1-\zeta^2})}{\zeta t_s} \tag{5.6}$$

This described appraoch delivered follwoing results:

	Results
ω_{RM}	20.2845
ζ	0.7887

Tabelle 5.2: Obtained analytical results

5.3 Weighting Filter Selection

An essential step in the design of robust control is it to compute or define the weights with which the controller will be created. The lecturer has given the function of the weights, but the parameter had to be determined. The functions for the weight and the bandwidth frequency are given as:

$$W_i = \frac{M_{h,i}s - \omega_i^{,} M_{l,i}}{s + \omega_i^{,}} \tag{5.7}$$

$$\omega_i^{,} = \left(\sqrt{\frac{M_{l,i}^2 + k_i^2}{k_i^2 - M_{l,i}^2}} \right) \omega_i \tag{5.8}$$

5.3.1 Approach in MATLAB

To determine the needed parameter in MATLAB, it was first needed to define the sensitivity function and the control times sensitivity. Both functions have later on used to receive the two needed Weights. The sensitivity function has been defined as:

$$S = 1 - T_{RM} \tag{5.9}$$

The control times sensitivity function has been defined as:

$$KS = T_{RM} \tag{5.10}$$

For both system has been the high and low-frequency gains computed by using the MATLAB function 'bode'. The also needed gain crossover frequency has been determined by reading the bode plotes at the gain of -3 dB. Like in the assignment paper suggested has been ki defined as 0.707.

5.3.2 Results

Target Transfer Function

The sigma function plots of the computed target transfer functions are displayed in figure 5.4.

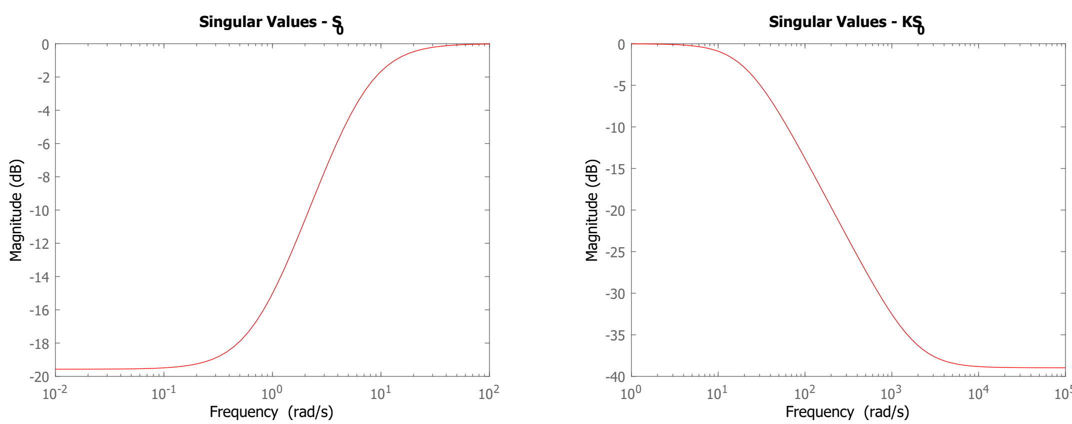

Figure 5.4: The sigma function plots of the computed target transfer functions

Singular Values

In figure 5.5 the relevant sigma function responses are shown.

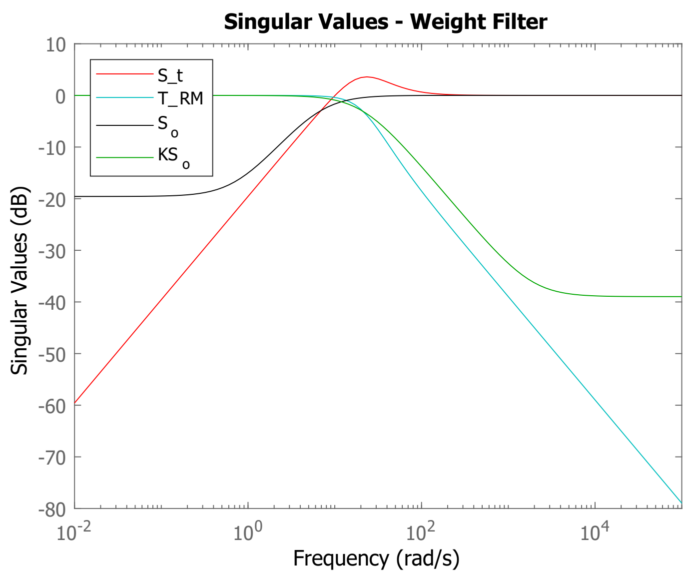

Figure 5.5: Sigma Function Responses of relevant Dynamics

5.4 Synthesis Block Diagram

After computing the relevant weights, the system had to be defined in a given standard form. The form is given in figure 5.6.

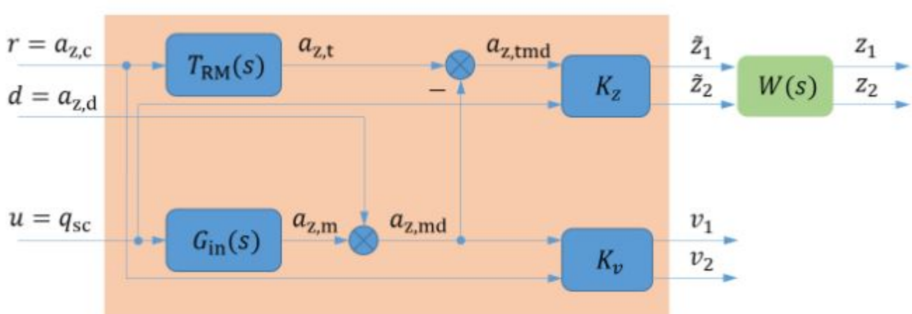

Figure 5.6: Controller Standard Form

5.4.1 Approach in MATLAB

In order to create the outer feedback loop of the uncertain plant dynamics, the 'SS()' and 'CONNECT()' functions of the control toolbox has been utilised. The functions allow it to connect the prior defined state space blocks properly and to allow an analysis of the assembly. As in figure 5.6 how the system shall be connected.

5.4.2 Results

The computed transfer functions have been extracted and are now presented in the following.

Transfer Function: $\widetilde{P}(s)$

$$T_{\widetilde{P}(s)} = \begin{bmatrix} z_1 \\ z_2 \\ v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \frac{-11.25s+411.5}{s^2+32s+411.5} & -1 & \frac{5431s^2-636s+7.239e06}{s^4+211.3s^3+2.307e04s^2+5.779e05s+7.239e06} \\ 0 & 1 & 0 \\ 0 & 1 & \frac{-5431s^2+636s+7.239e06}{s^4+211.3s^3+2.307e04s^2+5.779e05s+7.239e06} \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} r \\ d \\ u \end{bmatrix}$$

Transfer Function: $P(s)$

$$T_{P(s)} = \begin{bmatrix} z_1 \\ z_2 \\ v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \frac{-11.24s^2+333.4s+2837}{s^3+32.72s^2+434.6s+298.1} & \frac{-0.9992s-6.894}{s+0.7244} & \frac{5426s^3+3.681e04s^2-7.237e06s-4.991e07}{s^5+212s^4+2.323e04s^3+5.946e05s^2+7.657e06s+5.244e0} \\ 0 & \frac{88.82s-1847}{s+1847} & 0 \\ 0 & 1 & \frac{-5431s^2+636s+7.239e06}{s^4+211.3s^3+2.307e04s^2+5.779e05s+7.239e06} \\ 1 & 0 & 0 \end{bmatrix}$$

5.4.3 Controller Choice

The given controller standard form allows the control system to validate if the system is acting as it should be by comparing it to the reference model. The reference behaviour is compared with the reference model. If there is a disparency, this value will be interpreted as an undesired uncertainty error which is feed into the weight matrix. In general, it can be said that the system is splitting the problem in a part in which the control system deals directly with the target values and in a part in which the uncertainties are handled.

5.5 Controller Synthesis

5.5.1 Approach in MATLAB

In order to create a controller synthesis H_∞ontroller has been designed. The controller has been computed by utilising the 'HINFSYN' library function of the MATLAB robust control toolbox.

5.5.2 Results

Controller: $K_{FO}(s)$

The obtained controller is defined as:

$$K_{FO}(s) = \begin{bmatrix} \begin{pmatrix} 123 & 1 \\ 3\bar{1}23 & 3123 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 3123 \\ 132 & 3123 & 312 \end{pmatrix} \\ \begin{pmatrix} 132 & 0 \\ 0 & 0 \\ 3123 & 0 \\ 0 & 312 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 3123 \\ 0 & 0 & 3123 \\ 0 & 0 & 0 \end{pmatrix} \end{bmatrix}$$

Controller Decomposition

It was required to obtain the controllers K_{dr} and K_{cf} from the newly defined H_∞ controller. With an analysis of figure, 5.9 is it possible to understand that the System has 4 outputs. The z_1 and z_2 outputs are used to handle the uncertainties of the system by enhancing the signal with the defined weights. While v_1 and v_2 will be connected to the K_{F0} controller. These output values can be related to the in Figure 5.8 defined parameter of $a_{z,md}$ and $a_{z,c}$.

Since K_{dr} is a unity matrix and don't change the received values of $a_{z,c}$ and a_{ds} . These paramter are directly feet into the K_{F0} controller. As priorly defined are these parameters also used for the in figure 5.8 defined feedback system. Therefore it can be defined that the K_{F0} component which is handling $a_{z,c}$ is equal to K_{cf} and that the K_{F0} component which is handling $a_{z,md}$ is equal to K_{dr}.

Controller Order $K_{FO}(s)$

The minimal controller order has been computed by utilising the 'MINREAL' MATLAB library function. The order without the order reduction has been computed by using the 'ORDER' MATLAB library function and it was 8. The minreal function had no influences on the systems order.

Weigthed Closed Loop Transfer Function

Tee weigthed closed loop transfer function of the system is:

$$T_{\widetilde{P}(s)} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} \frac{-11.24s^2+333.5s+2837}{s^3+32.72s^2+434.6s+298.1} & \frac{-0.9992s-6.894}{s+0.7244} \\ 0 & \frac{88.82s+18475}{s+1847} \end{bmatrix} \begin{bmatrix} r \\ d \end{bmatrix}$$

5.5.3 Singular Value Analysis

For a further understanding of the system singular values for the unweigthed closed loop and the corresponding weighting function has been computed and displayed in the figure 5.7.

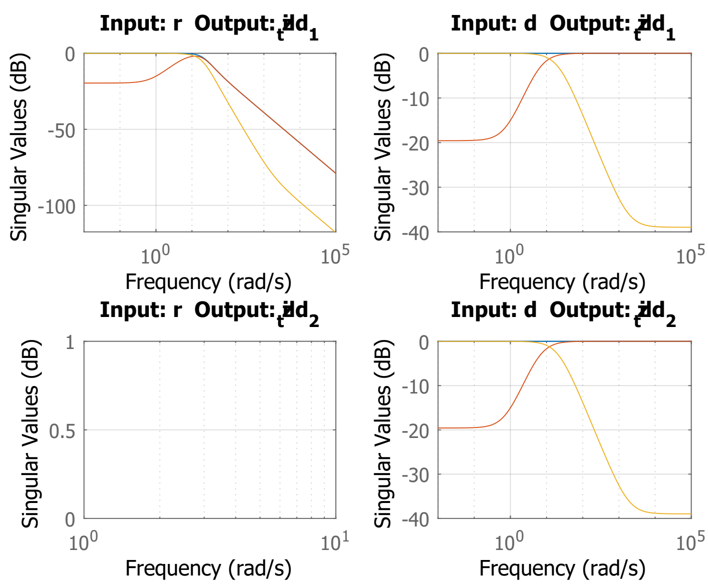

Figure 5.7: 2x2 Singular Values Plot

5.6 Controller Implementation

In the next step, it is required to convert the controller scheme in figure 5.8

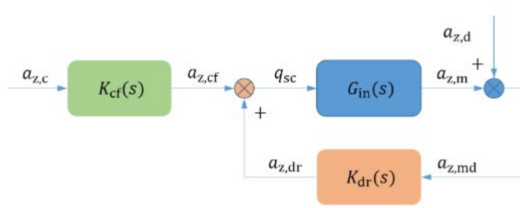

Figure 5.8: Prior defined outer closed loop

Into a system given in figure 5.9.

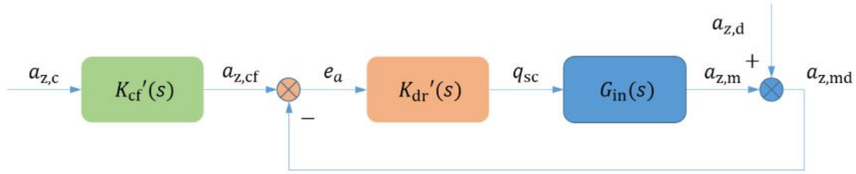

Figure 5.9: Desired controller implementation

5.6.1 Analytical Approach

In order to determine the gains $K^{'}_{cf}$ and $K^{'}_{dr}$, an analytical approach has been used. The classical control engineering theory delivers tools for creating equivalent control systems block structures.The result of the equivalent system is given in figure 5.10.

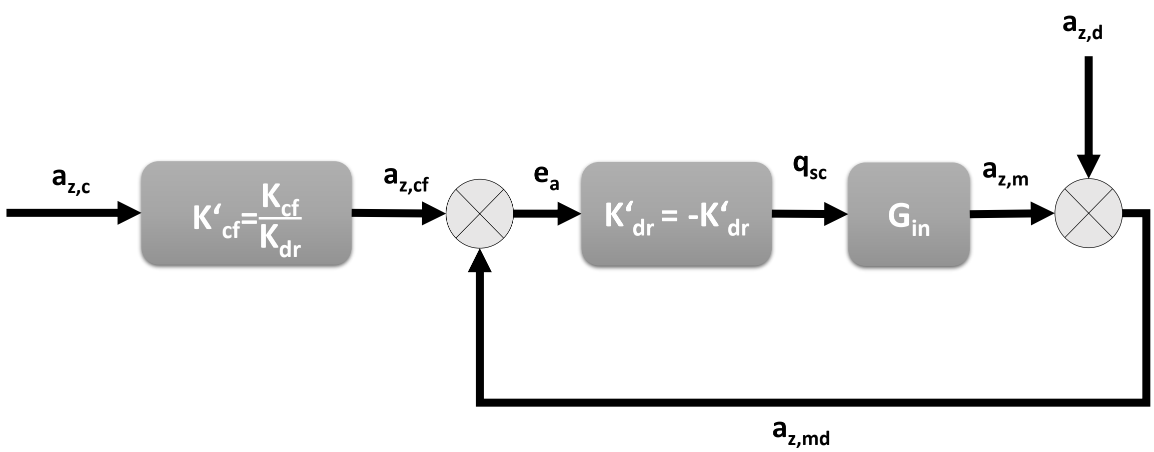

Figure 5.10: Analytical Solution for given problem

5.6.2 Controller Characteristic

The controller has been further investigated for its characteristics.

Controller Order

The controller order has been computed by utilising the 'ORDER' MATLAB library function, This function delivered the result that the order of the controller is 8.

Poles and Zeros

The poles and zeros of the controller system are given in the figure 5.11.

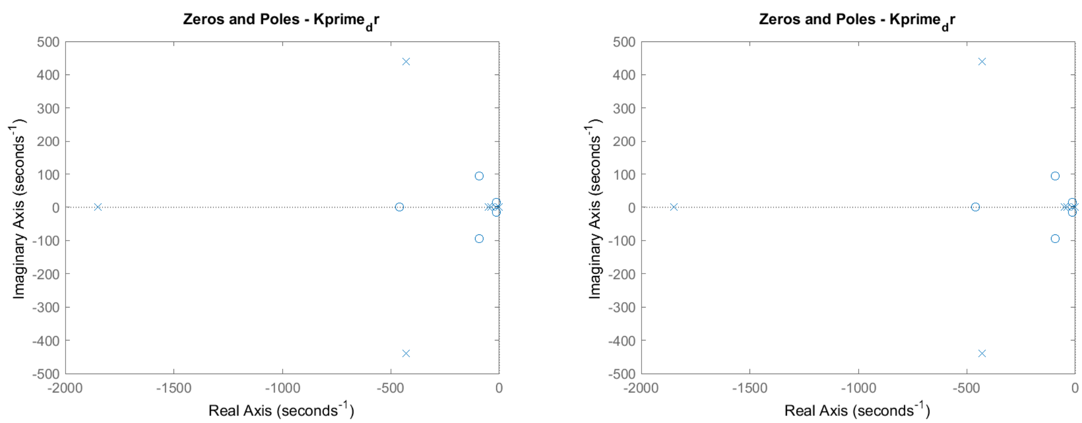

Figure 5.11: Z-Plan Plot of both obtained controllers

High Frequency Dynamic

In order to obtain the high frequency dynamic a bode plot has been created.

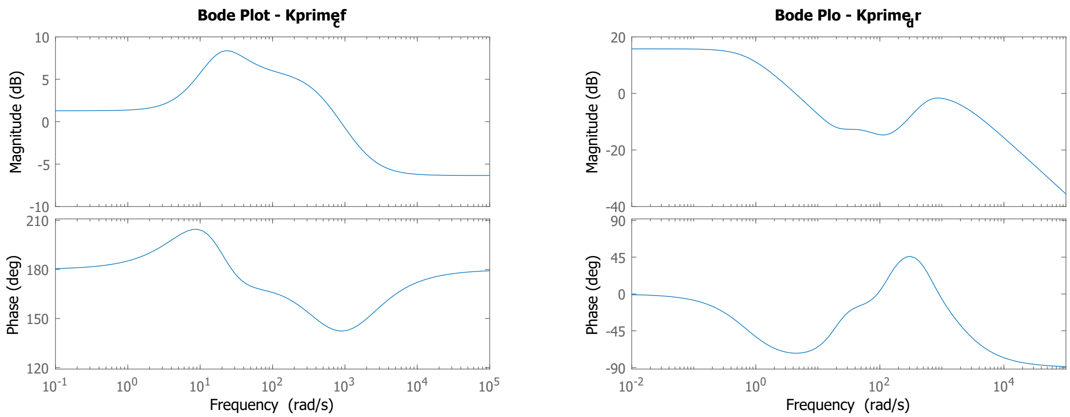

Figure 5.12: Bode Plot of both obtained controllers

5.6.3 Controller Reduction

The minimal controller order has been computed by utilising the 'REDUCE' MATLAB library function. The order without the order reduction has been computed by using the 'ORDER' MATLAB library function and it was 8. The 'REDUCE' has provided a diagramm which allows to see the influences of an order reduction to the error. It shows that the higher the reduction is, the bigger gets the error. Therefore trad-off as been made and the third order reduction has been utilised.

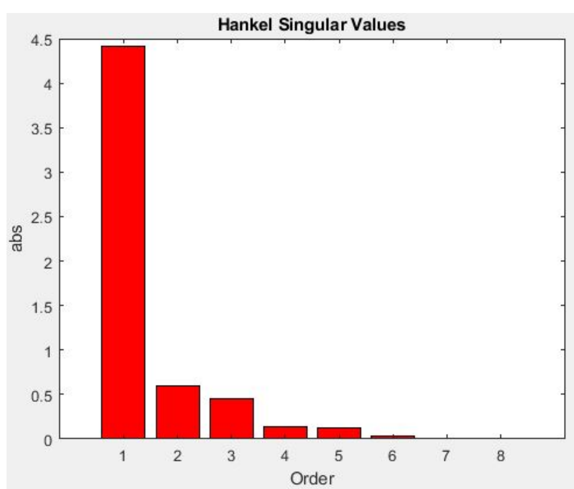

Figure 5.13: Sigma Function Responses of relevant Dynamics

5.6.4 Comparision of Full Order and Reduced Controllers

The native and the reduced systems have been compared by utilising the SIGMA() function of Matlab. The results are shown in figure 5.14.

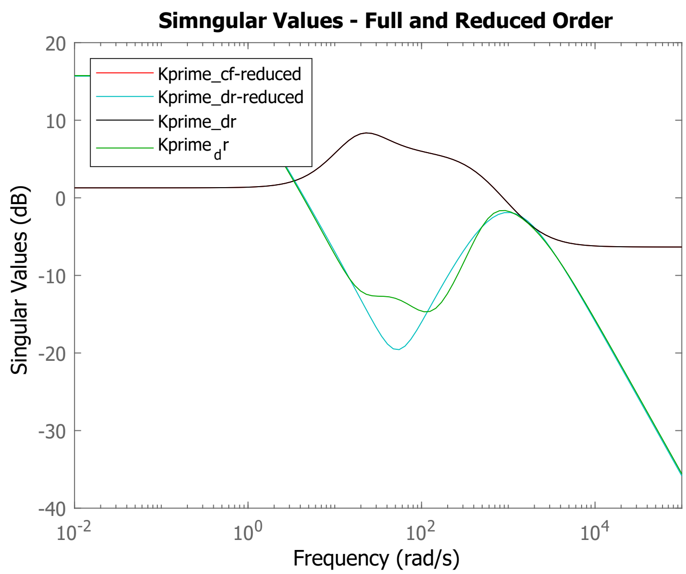

Figure 5.14: Sigma function comparision of reduced and native system

5.7 Controller Simulation

For a final validation and perfomance analysis of the designed system the system has been duplicated in the Simulink environment.

5.7.1 Approach im Simulink

For the control simulation, the created system has been replicated in Simulink. The given global system is displayed in figure 5.15.

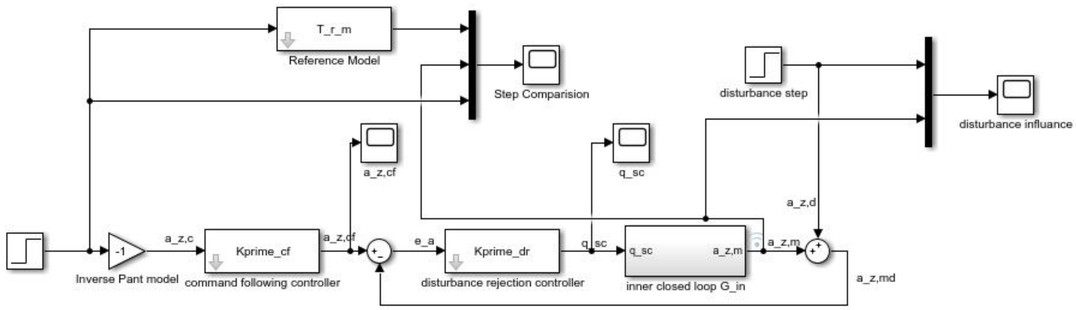

Figure 5.15: Overall Control System in Simulink

The plant model with its actator, airframe dynamics, gains and sensor blocks is defined in a subsystem, which is shown in figure 5.16.

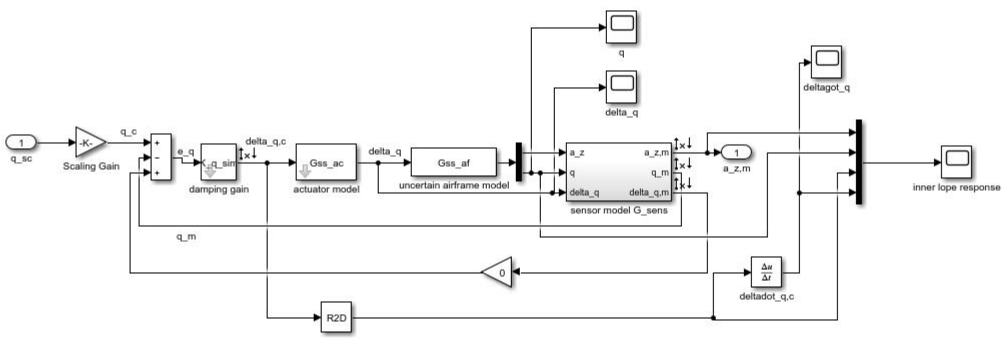

Figure 5.16: Inner Closed Loop System in Simulink

5.7.2 Response Behaviour and Disturbance Rejection Performance

A step input of 1 (regarding to1g load factor) has been applied to the system. The outputs for the normal acceleration response, the pitch rate q, the control signal δ(in degrees) and the time derivative of the control signal $\dot{\delta}$. In order to analyse the disturband performance, a step input of 0.1 (regarding 0.1g load factor) has been applied to the system after 1 second.

Response: a_z

In figure 5.17 can be the output of a_z seen. The system is follwoing, after a little undershoot, the reference properly and reaches a settling in around 0.2 seconds. After a little overshoot it converges to the given reference value. After 1 second a disturbance is given to the system. This can be seen in the jump after second, but also for this new situation is the system taking the disturbance value as an offset and maintains a new value.

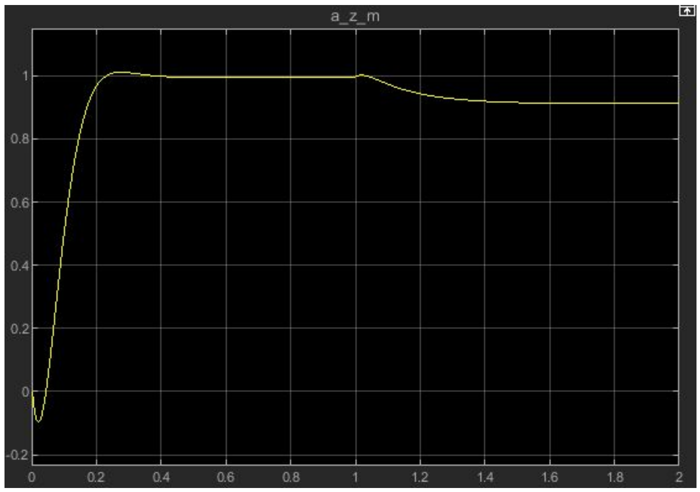

Figure 5.17: Step response of a_z

Response: q

Also q is following the reference (figure 5.18)

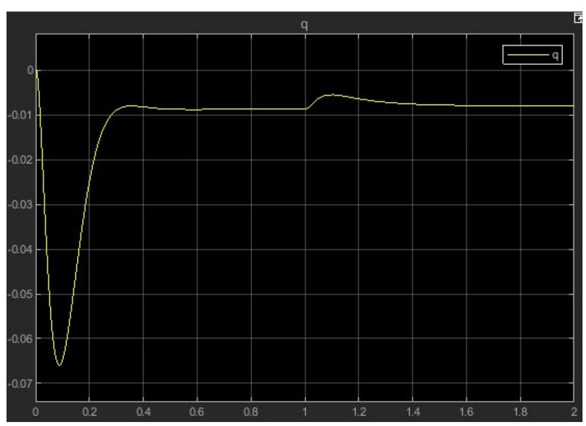

Figure 5.18: Step response of q

Response: $\delta_{q,c}$

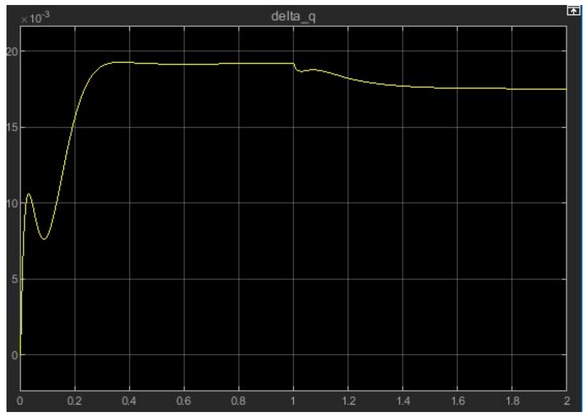

Figure 5.19: Step response of $\delta_{q,c}$

Response: $\dot{\delta_{qc}}$

$\dot{\delta_{qc}}$ shows in figure 5.20 exactly two spikes, which can be related to the both step inputs.

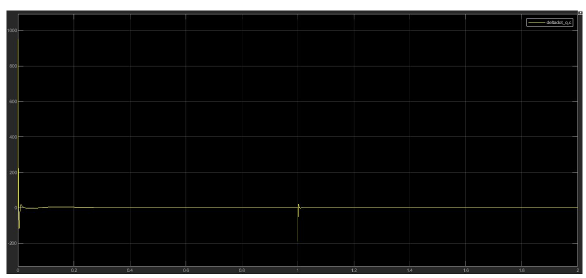

Figure 5.20: Step response of $\dot{\delta_{qc}}$

5.7.3 Comparison of Reference Model and Plant

A step input of 1 (regarding to1g load factor) has been applied to the system. The outputs for the normal acceleration response, the pitch rate q, the control signal δ(in degrees) and the time derivative of the control signal $\dot{\delta}$. In order to analyse the disturband performance, a step input of 0.1 (regarding 0.1g load factor) has been applied to the system after 1 second.

Chapter 6

Robust_Analysis

6.1 Verfication of Gain and Phase margins

6.1.1 Approach in Simulink

In order to use Simulink to verify the gain and phase margins of the resulting SISO loops the Simulink Control Designer have been utilised. The tool is able to analyse the system from point to point by opening the loop at the input of the actuator and the output of each sensor.

6.1.2 Verfication Step with Simulink Control Designer

The verfication step in Matlab delivered in figure 6.1 the bode plots for the three measured parameters.

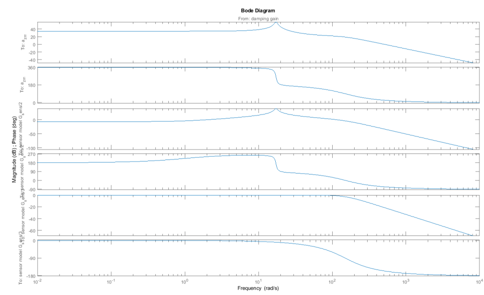

Figure 6.1: Bode Plots of open loop (from actuator to the sensors)

6.2 Advanced Robustness Analysis

6.2.1 Approach in Simulink

To use Simulink to analyse the system the robustness of the system, a new Simulink model had to be created. The new control system structure is using the M and delta matrices which were computed with lftdata in a previous step. The Simulink model than the load to Matlab to apply further functions from the robust control library. The Simulink model is shown in figure 6.2.

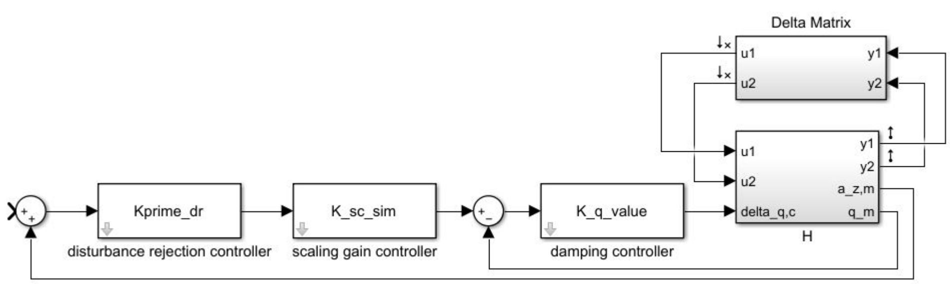

Figure 6.2: Simulink Model for Robust Analysis

6.2.2 Control System Structure

In the given control block diagram for the robust analysis, which is shown in figure 6.3, no feedforward controller is implemented. This can be explained with the fact that the feedforward controller has no influence on the robustness behaviour of the system The feedfordward controller is adapting the mean value of the output in way it is desired, but it is not able to reject disturbances or handle uncertainties.

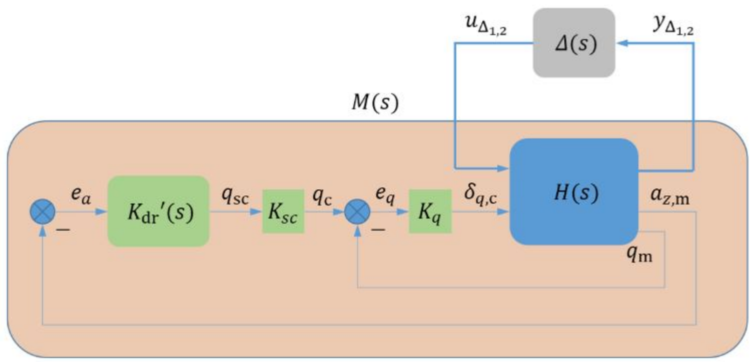

Figure 6.3: Given control system structure

6.2.3 Transferfunction

In order to receive a further understanding of the differences between the different dynamic characteristics of an uncertain and idealised system, both systems had been compared in different methodologies. Stabal system are on a fixed place, while the zeros and poles of the uncertain system are distributed in a certain area.The system dynamic behaviour is in both cases stable, but the system has a non minimum phase zero, which has it's orgin from the airframe system.

$$M(s) = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \frac{169.8s^6+5.443e05s^5+4.208e08s^4+1.567e11s^3+1.161e13s^2+2.384e14s+3.847e14}{s^8+3222s^7+2.538e06s^6+9.73e08s^5+8.868e10s^4+3.08e12s^3+4.636e13s^2+2.902e14s-1.492e15} & \frac{84}{s^8+3222} \\ \frac{14.2s^7+3.972e04s^6+3.091e07s^5+1.216e10s^4+8.707e11s^3+1.471e13s^2-8.104e13s-1.419e15}{s^8+3222s^7+2.538e06s^6+9.73e08s^5+8.868e10s^4+3.08e12s^3+4.636e13s^2+2.902e14s-1.492e15} & \frac{7.064s^7}{s^8+3222} \end{bmatrix}$$

6.2.4 Mu-Analysis

The applied Mu-analysis has delivered the results which can be seen in figure 6.4.

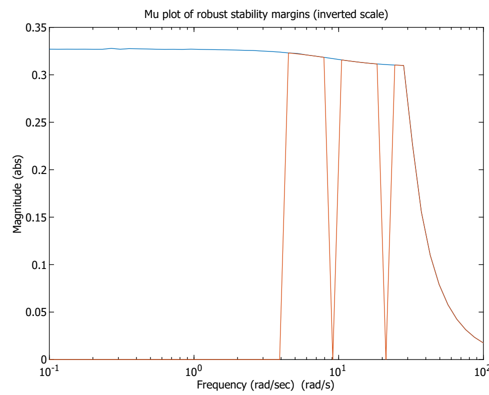

Figure 6.4: Mu-Plot of System

Bibliography

[1] D. K. Arrowsmith and C. M. Place. *"The Linearization Theorem" - Dynamical Systems: Differential Equations, Maps, and Chaotic Behaviour*. London: Chapman & Hall, 1992.

[2] J. Autenrieb. The mathematical modeling, trimming and linearisation of an transsonic missile and a further control law development. Master's thesis, Cranfield University, 2018.

[3] M. V. Cook. *Flight dynamics principles: a linear systems approach to aircraft stability and control*. Butterworth-Heinemann, 2012.

[4] B. N. N. JOHN E. CASHMAN, BRIAN D. KELLY. Operational use of angle of attack. *Aero magazine*, 2010.

[5] G. LAJOIE. The hartman-grobman theorem and the equivalence of linear systems. Online, 2006.

[6] R. C. Nelson. *Flight stability and automatic control*, volume 2. WCB/McGraw Hill New York, 1998.

[7] N. Nise. *Control Systems Engineering*. John Wiley & Sons, 1992.

[8] P. H. Zipfel. *Modeling and Simulation of Aerospace Vehicle Dynamics*. AIAA, 2000.