Confidence metrics analysis of a fixed-wing UAV

A THESIS SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL OF THE UNIVERSITY OF MINNESOTA BY

Janos Polgar

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE

Gary J. Balas

Peter J. Seiler

December, 2013

Acknowledgements

I would like to thank my advisor, *Gary Balas*, that he believed in me, gave me the opprortunity to participate in this project, and study at the University of Minnesota.

Pete Seiler proved me, how can someone be excellent in theory and in application as well. Moreover, he took over the leading position on my best teachers list. I assume, forever...

Andrei Dorobantu introduced me to the UAV simulation and flight dynamics.

David Escobar Sanabria presented me a next level in programming.

Brian Taylor and *Danny Christ* turned the C codes into flight experiment results.

Will Thorson made me laugh all the time in the lab.

Harald Pfifer, Claudia Moreno, Paul Freeman, Daniel Showers, Jen Annoni, Shu Wang, Bin Hu, Masanori Honda and Aditya Kotikalpudi made an inspiring and valuable environment in the lab.

Last but not least, I would like to thank my *family* their support and love.

Contents

List of Tables

List of Figures

Nomenclature

a_x, a_y, a_z	body axis accelerations $\left(\dfrac{\mathrm{m}}{\mathrm{s}^2}\right)$
h	altitude (m)
p, q, r	body axis angular rates $\left(\dfrac{\mathrm{rad}}{\mathrm{s}}\right)$
u, v, w	body axis velocities $\left(\dfrac{\mathrm{m}}{\mathrm{s}}\right)$
T_s	sampling time (s)
v_e, v_n	east and north velocity components $\left(\dfrac{\mathrm{m}}{\mathrm{s}}\right)$
V	airspeed $\left(\dfrac{\mathrm{m}}{\mathrm{s}}\right)$
α	angle of attack (rad)
β	sideslip angle (rad)
δ_{ail}	aileron deflection (rad)
δ_{ele}	elevator deflection (rad)
δ_{rud}	rudder deflection (rad)
δ_{thr}	throttle command (0-1)
Ω	propeller rotation speed
ϕ, θ, ψ	attitude angles (rad)
ψ_g	ground track angle (rad)

Chapter 1

Introduction

Uninhabited aerial vehicles (UAVs) are becoming popular in the development process of full scale aircrafts and as research platforms. Due to their complexity they provide development and test environments for a wide range of applications. Supporting research projects in safety critical systems, classes, the University of Minnesota Department of Aerospace Engineering and Mechanics have been developing a low-cost UAV research facility. This facility includes models of a family of fixed wing airframes, controllers, a diverse set of guidance algorithms. A flight software is written which implements an autopilot system, including the aforementioned algorithms, and provides datalogging. The software package is equipped with tools to evaluate flight test results.

The model of any plant is never 100% accurate. There are always differences between the real system and the dynamical model of it. Uncertainties can be introduced into the model, which are trying to capture uncertainty in model parameters and unmodeled dynamics [12]. Even though the aircraft model in the package is fairly accurate, it is interesting to investigate 'how good' the model is, i.e. how robust the model in the closed loop is against uncertainties.

Earlier work in this project mainly focused on plant modeling and controller design [2, 7]. Extensive controller analysis, however, has not performed yet, what motivates the work behind this thesis.

The thesis is organized as follows. Chapter 2 gives an overview about the aircraft model, as well as the available controllers. Uncertainty models and linear robustness analysis results are presented in Chapter 3. Chapter 4 presents tools for robustness analysis, as well as results on the nonlinear model of a UAV. Flight tests were performed to validate the effects of perturbations in the closed loop. Appendices give a brief overview about the theoretical background of robustness analysis tool, contain numerical data as well as flight test results.

Chapter 2

UMN UAV Simulation

The UMN UAV research group has an extensive simulation environment for modeling and control of different vehicles, available online as an open source software environment [11]. [3] describes the fleet of airframes, the sensors mounted on them as well as the flight software.

The 6 degree of freedom (DOF) model of a fixed wing aircraft was implemented in Simulink. The force and moment calculations partly depend on the Aerospace Blockset of Simulink. Actuator and sensor models are also included into the loop. Thank to the modular setup of the simulation, the aircraft parameteres are easily interchangeable, and do not require major modifications in the sofware. The control law can be chosen from a set of avaliable controllers implemented in either Simulink or in C. Sections 2.2.1 and 2.2.2 contain detailed information on the controllers.

The simulation enviroment has three well defined parts as described in the sequel. The *nonlinear simulation* contains the 6 DOF rigid body aircraft model, as well provides tools for trimming and linearizing the model around the specified trim point. The *software-in-the-loop* simulation provides the closed loop configuration in feedback with a control law. The *hardware-in-the-loop* simulation is a transition toward the real airframe, as it contains the flight computer with the control law implemented in C code. It acquisites the sensor data from the nonlinear Simulink model and provides the actuator commands toward the Simulink model. Before conducting flight test with an algorithm, SIL and HIL sims are used to debug the newly written code and evaluate the performance of control or guidance law.

The software package contains also the flight software written in C, which uses the aforementioned controller implementations, therefore there is no need to develop a controller code to run simulations and another one to conduct flight tests.

Linearized models are used to perform controller analysis. In the linearization process it is assumed that the longitudinal and lateral/directional dynamics are completely decoupled, therefore one obtains two state space models. The longitudinal model has the states: $\begin{bmatrix} u & w & q & \theta & \Omega \end{bmatrix}^T$. Outputs are defined as $\begin{bmatrix} V & \alpha & q & \theta & h & a_x & a_z \end{bmatrix}^T$.

The lateral/directional model has a state space as: $\begin{bmatrix} v & p & r & \phi & \psi \end{bmatrix}^T$. States $\begin{bmatrix} \beta & p & r & \phi & \psi \end{bmatrix}^T$ were chosen as outputs.

2.1 Plant dynamics

This section has no intention to draw a complete picture about the model of the Ultrastick 25e. [4, 5] detail the system indentification process and the obtained model. It adresses, however, a couple of model properties, which are important in further sections.

The identification of linear model parameters consisted of two steps. First, a smaller airframe underwent wind tunnel testing, which was supposed to be a scaled version of the 25e. In fact it is not a scaled version of it, only a very smimilar, but smaller airframe. The model obtained from wind tunnel experiments was used as a baseline for the second step: parameter fitting using flight experiment data. Significant improvements on the model have been achieved by this step.

The results in [5] showed that the parameter identification in the yaw axis was the least accurate among the three axes. The main reason behind this is that the aileron to yaw rate transfer has a low gain, which results a lower signal to noise ratio. This makes the identification more challenging.

2.2 Software-in-the-loop simulation

The closed control loop in the SIL simulation can be seen in Figure 2.1.

The control law has a two-layer structure: the outer loop maintains velocity, altitude and guidance commands. The inner loops are to track roll and pitch attitude reference signals determined by the outer loop.

The outer-loop controller performance has to fulfil requirements such as no overshoot in the

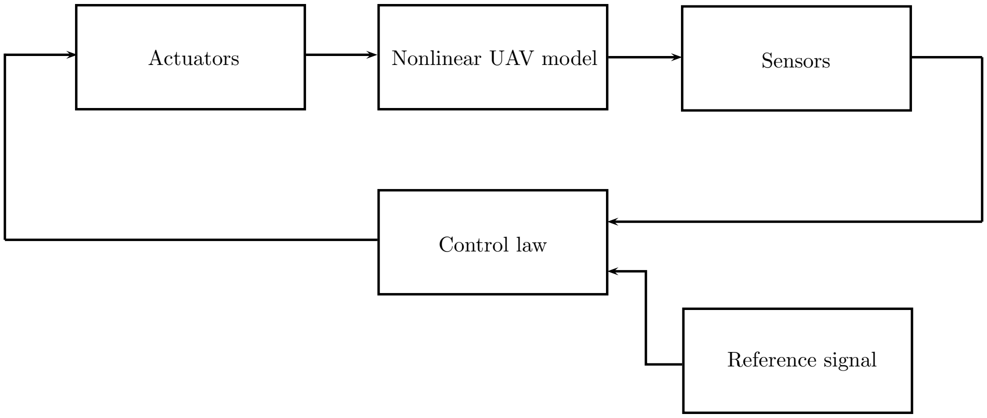


Figure 2.1: SIL simulation loop

ground track angle, altitude accuracy with ±5 m and maintain airspeed within ±2 m/s. Several design goals were introduced against the inner loop performance: the closed loop rise time should be less than 1 sec, the overshoot has to be smaller than 5%, as well as a 6 dB gain margin and ±45° phase margin were required. The controllers were tuned using linearized model of the aircraft.

Two inner loop controllers are available: the baseline design in PI structure as well as an LQR controller. The detailed description of both the outer- and inner-loop controllers is given in the sequel.

2.2.1 Outer-loop controller

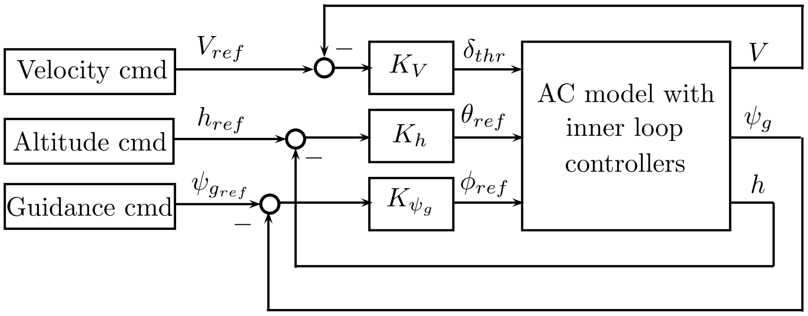


Figure 2.2: Outer loop controller

Figure 2.2 represents the outer loop controller architecture. The velocity, altitude and guidance commands (V_{ref}, h_{ref}, $\psi_{g_{ref}}$) are determined in the guidance code part of the package. ψ_g denotes the ground track angle calculated as $\psi_g = \arctan \frac{v_e}{v_n}$, where v_e and v_n are east and north velocity components, respectively. Proportional-integral blocks give the inputs for the inner loop controllers (δ_{thr}, θ_{ref}) in case of the throttle and altitude controller, while a proportional gain was chosen as the heading controller (ϕ_{ref}). Anti windup logic ensures that the throttle and altitude commands remain inside the authorized working region of actuators (block is not shown in Figure 2.2). The throttle commands go directly to the aircraft model without any modification by the inner loop.

Roll angle reference is constrained at ±45°, while pitch angle has to remain between ±20°. Throttle command is limited between the range of 0 and 1.

2.2.2 Inner-loop controller

Separated controllers are responsible for tracking commands in the pitch and lateral/directional axes. Figure 2.3 shows the structure of longitudinal controller. As mentioned before, throttle command (δ_{thr}) provided by the outer loop passes the inner loop without modification. A proportional pitch rate gain (K_q) increases the damping, while a proportional-integral term (K_θ) is responsible for the attitude tracking in the feedback of θ. The pitch angle reference signal (θ_{ref}) is calculated in the outer loop altitude controller.

Anti windup logic is implemented both in the longitudinal and lateral/directional controller realizations in order to avoid actuator saturation.

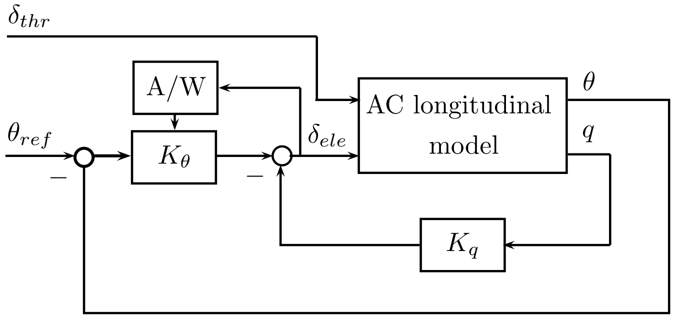


Figure 2.3: Baseline longitudinal axis controller

The lateral/directional controller was designed as shown in Figure 2.4. Three output signals are in feedback: yaw rate (r), roll rate (p) as well as roll angle (ϕ). Proportional gains (K_r, K_p, respectively) are in the loops of rate signals to increase the yaw rate and roll rate damping. The bank angle reference tracking is achieved using only lateral signals (p and ϕ). As a consequence, the rudder deflection command is calculated using only a yaw damper, i.e. a positive feedback through gain K_r. The aileron command signal is built using p and ϕ both in negative feedback. The outer loop guidance controller calculates the reference signal for this inner loop (ϕ_{ref}).

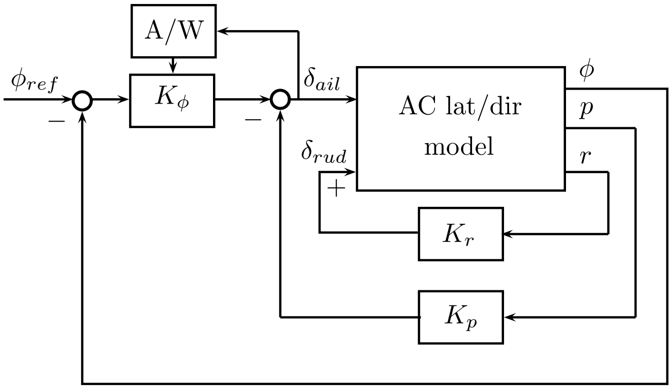


Figure 2.4: Baseline lateral/directional axis controller

The LQR controller has been developed only to handle the lateral/directional axes.

The block diagram is shown in Figure 2.5. In this setup, K_{00} and K_{10} represent integral gains, while all the other blocks denote proportional gains. Conditions are built in the code to take into account the actuator saturation.

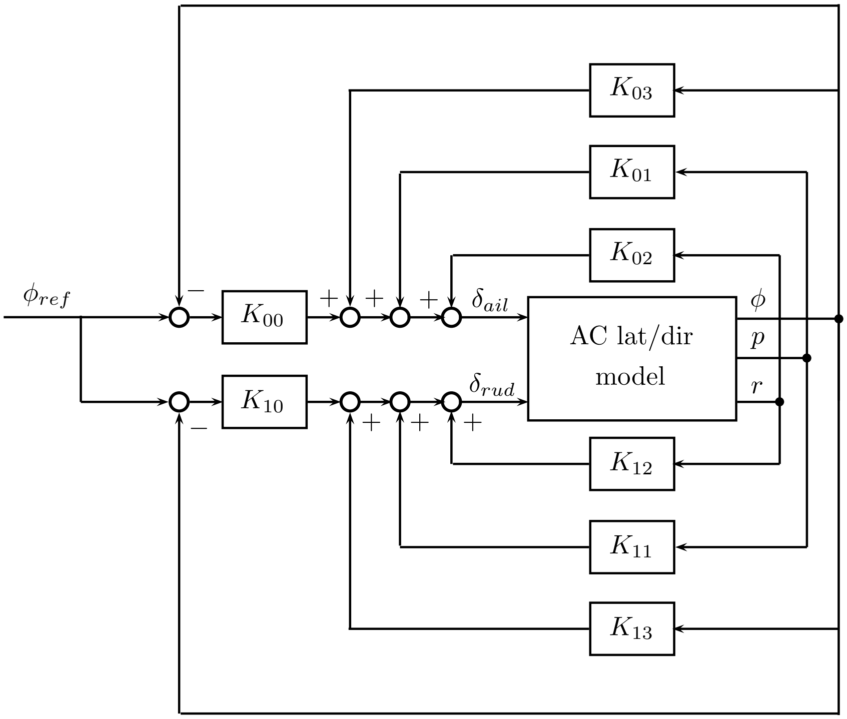


Figure 2.5: LQR lateral/directional axis controller

2.3 Inner loop responses

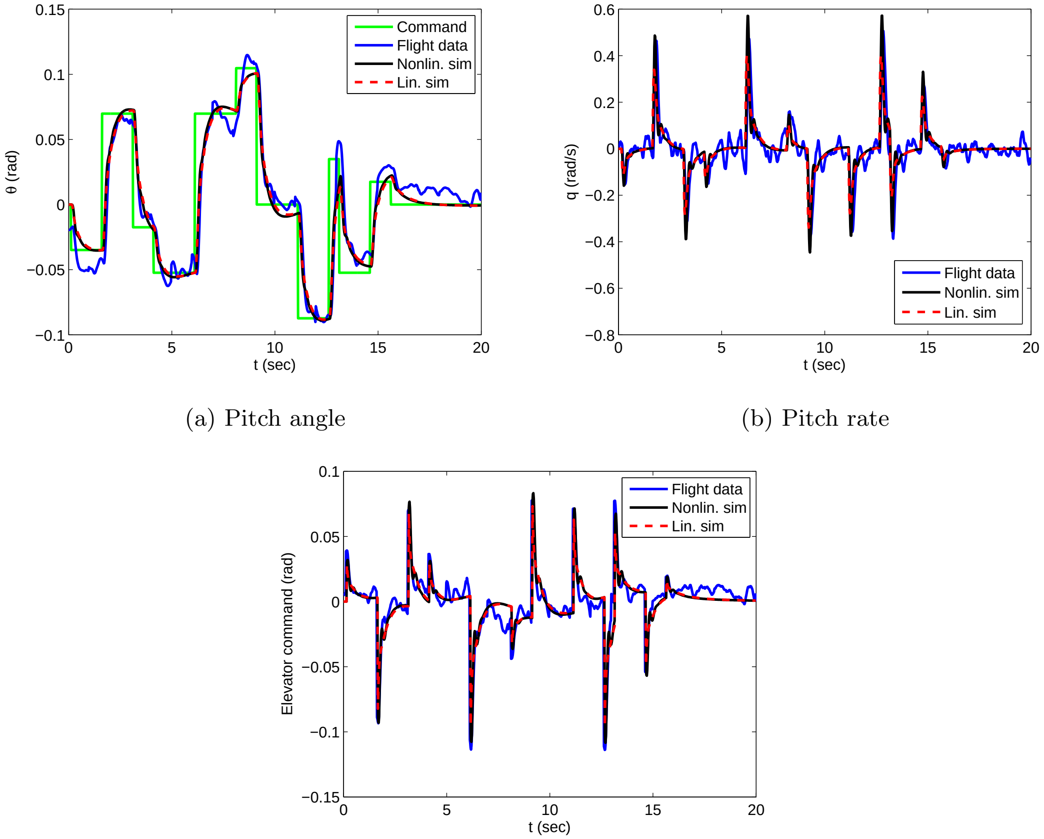
Time domain responses are presented in the sequel from simulation of closed loops. With the aid of the diagrams below one can compare the responses of the linearized dynamics as well as the nonlinear model with flight data.

The test flights were conducted with an Ultrastick 25e aircraft (nicknamed as Thor). The aim of the flights was to test the performance of the inner-loop controllers. The reference signals (θ_{ref} and ϕ_{ref}) were taken from the flight tests and applied to the models. Nither of simulation models include any noise source in order to provide easlily comparable results.

2.3.1 Longitudinal responses

Figure 2.6 shows the longitudinal responses due to a pitch angle reference signal depicted in Figure 2.6a. Analyzing the signal in Figure 2.6a one can conclude that the pitch controller fulfils the requirement on rise time as well as overshoot. The linear and nonlinear model responses do not differ significantly, however they do not match entirely with the flight test data due to disturbances (wing gusts, turbulence).

Pitch rate signals (Figure 2.6b) of the linear and nonlinear simulations are almost identical, although the peak values are less in case of the linear sim. The flight data peaks are between the linear and nonlinear responses regarding the magnitude. Both of them follow the trend of the signal taken from flight data, which, again, is disturbed by noise. One can conclude the same about the elevator command signals shown in Figure 2.6c.



(a) Pitch angle

(b) Pitch rate

(c) Elevator deflection

Figure 2.6: Longitudinal responses

Regarding the the other outputs of the longitudinal model, we can claim the followings. The linear and nonlinear closed loop systems produce the same signals, except high peaks, where the nonlinear response is slighlty greater. The flight data is noisier in case of all theses outputs. The tendency of accelaration signals is identical with the flight data, nevertheless the airspeed, angle of attack and altitude signals are slightly off compared to the experimental data.

2.3.2 Lateral/directional responses

A versatile roll angle reference signal was chosen to excite the dynamics of the aircraft at a wide frequency range. The same signal was applied both with baseline and LQR controllers. The baseline responses can be seen in Figure 2.7. The linear and nonlinear ϕ responses are identical in practice (Figure 2.7a). However, except some noise, they match with the flight data response as well. The roll rate responses of the linear and nonlinear systems are very similar, however the linear closed loop has a slightly smaller rise time. Unlike in the longitudinal axis, the flight data peaks are the highest in the lateral/directional axes. The peaks on the linear simulation, however, are still smaller than those in the nonlinear plots. These features can be observed on all the other response diagrams, except the yaw rate (Figure 2.7d). This controller also fulfils the requirements against rise time and overshoot.

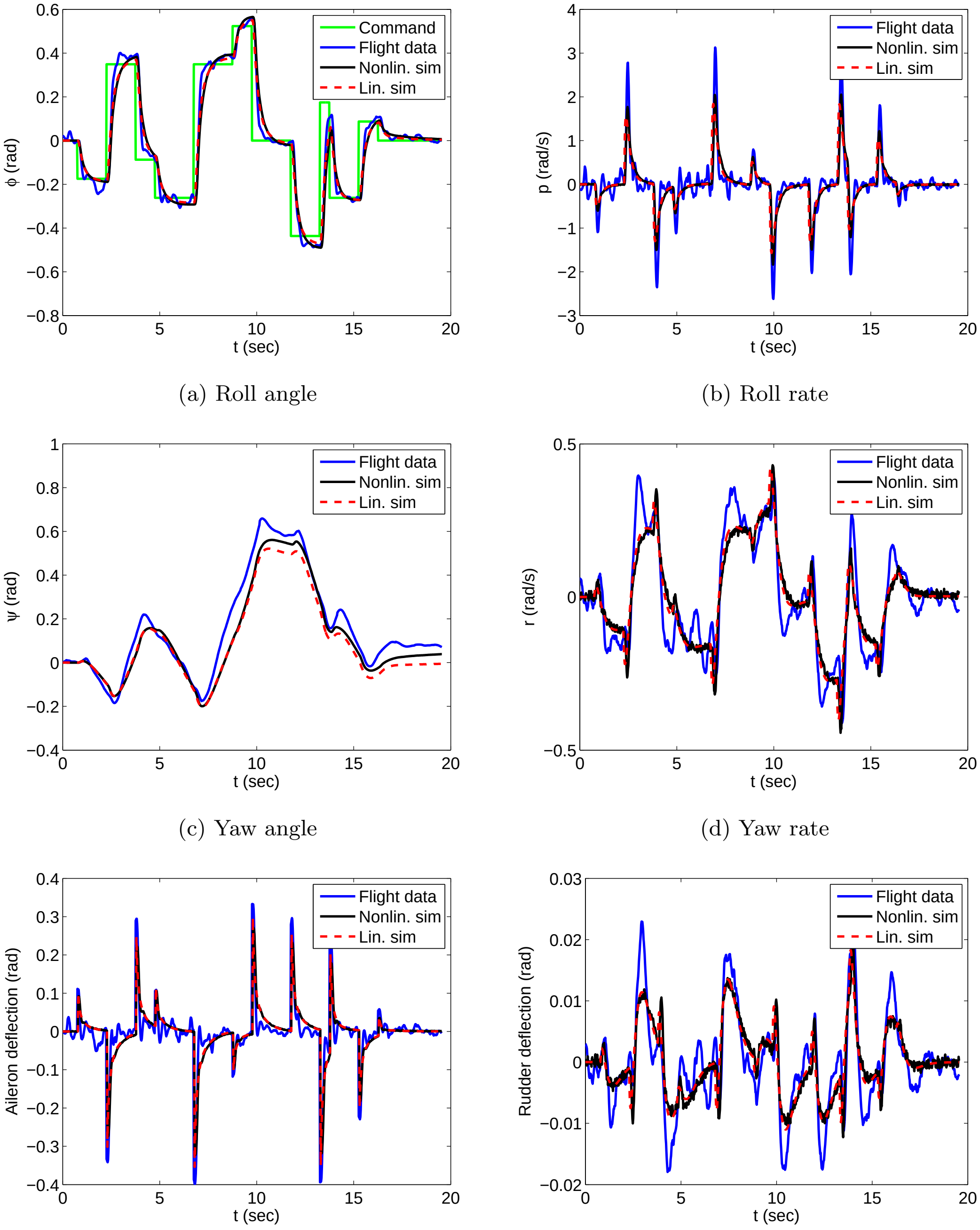
The yaw angle signals (Figure 2.7c) are identical by the third peak, however the linear ψ output become slighly smaller after the big amplitude peak. The yaw rate (Figure 2.7d) responses have similar characteristics as the roll rate signals, but they differ from the flight data. The latter one has some extra peaks on right after a significant change in the rate (due to a change in the reference signal).

The linear aileron deflection signal also represents the slighly quicker response time (Figure 2.7e). The interesting plot is the rudder command (Figure 2.7f), as it exhibits a behavior typical for a transfer function with a real right-half-plane zero. Due to a step in the roll angle reference, the rudder first moves to the opposite direction. Moreover, the simulation results have the biggest difference compared to the flight data.

The responses with LQR controller are collected in Figure 2.8. This controllers has a slightly greater rise time compared to the baseline design on the ϕ signal. The difference between the linear and nonlinear responses are visible at the peaks. The peaks on the roll rate signal have smaller magnitude and they are 'broader' in time, than the ones on the baseline response. The aileron deflection peaks have significantly smaller amplitude than those in the baseline controller. The linear and nonlinear yaw axis signals do not differ significantly.

From the plots below we can conclude that LQR controller is less agressive in the roll axis, than the baseline design, while provides similar responses in the yaw axis.

As a conclusion of the time domain signal analysis one can claim that the nonlinearities have only a small effect on the behaviour of the system. Therefore in the nonlinear analysis we can expect similar results as the linear gives.



(a) Roll angle

(b) Roll rate

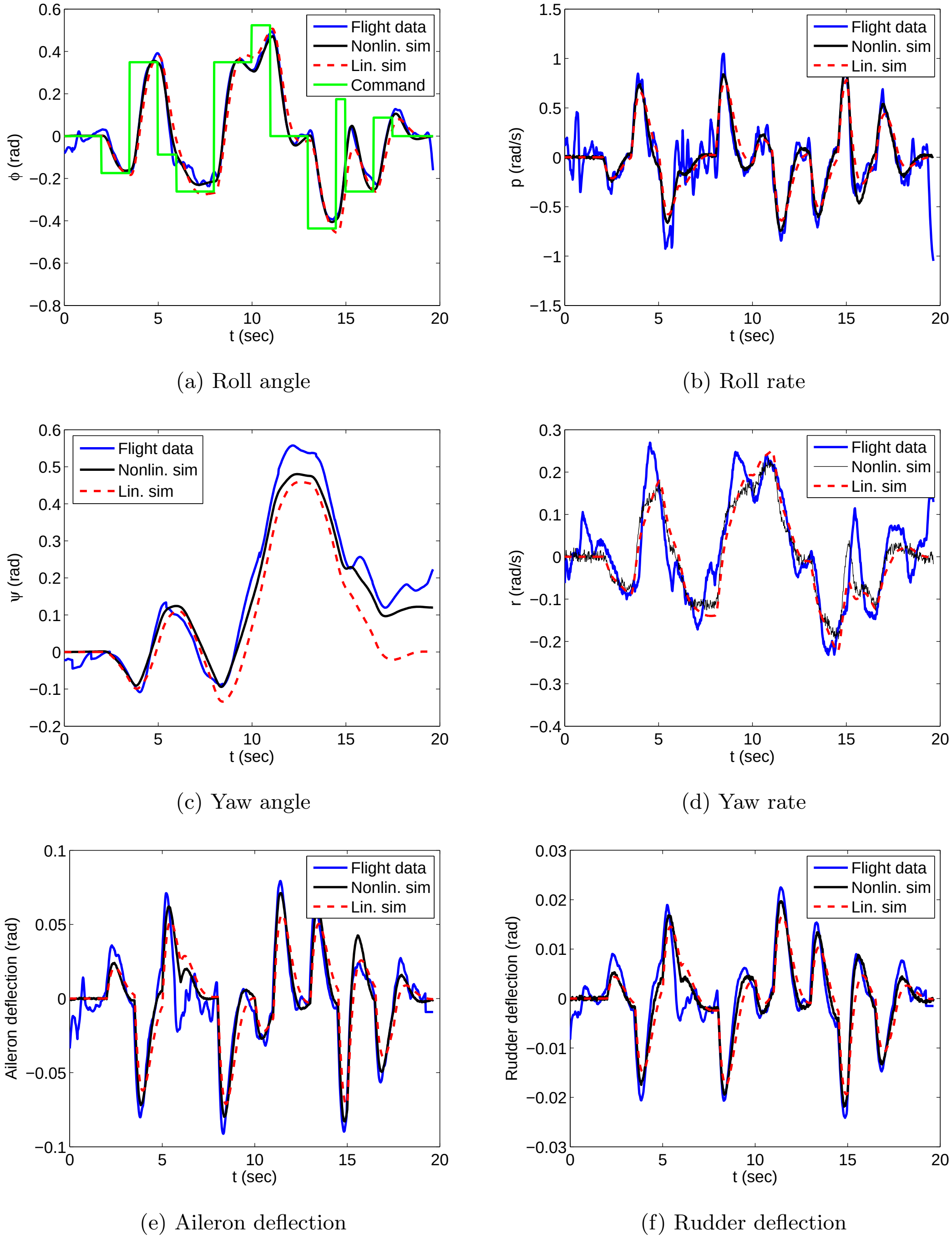
(c) Yaw angle

(d) Yaw rate

(e) Aileron deflection

(f) Rudder deflection

Figure 2.7: Lateral/directional responses with baseline controller



(a) Roll angle

(b) Roll rate

(c) Yaw angle

(d) Yaw rate

(e) Aileron deflection

(f) Rudder deflection

Figure 2.8: Lateral/directional responses with LQR controller

Chapter 3

Linear analysis

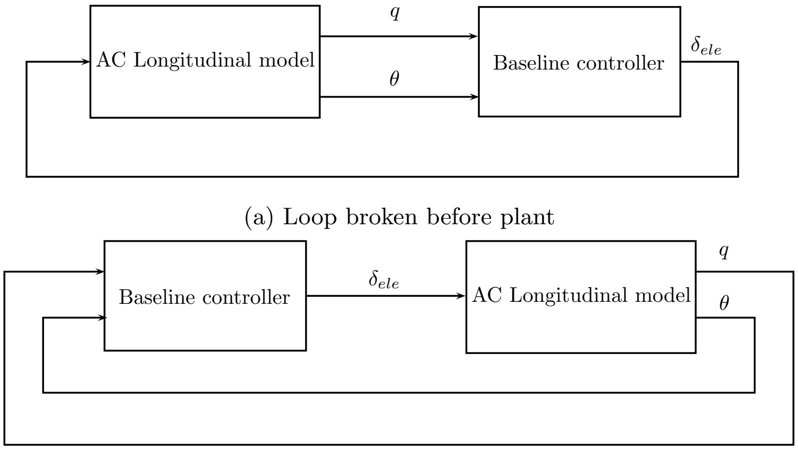
Classical margins as well as robustness analysis results are presented in this section. The analyzed systems were the linearized longitudinal and lateral/directional models with baseline or LQR controllers.

Robust stability, robust performance and worst case gain calculations give lower and upper bounds on the corresponding quantity at a certain frequency. Since in all calculations below these bounds differ only in the third digit after the decimal point, plotting of the lower bound is omitted to keep the figures more visible.

3.1 Longitudinal baseline controller

3.1.1 Classical margins

Gain, phase and time delay margins were computed in loop-at-a-time way. As it is well known, classical margins can be calculated only in a SISO system, therefore in the MIMO system one signal, either before or after the plant was broken (Figure 3.1). Only outputs in feedback were considered. With the aid of Matlab command `allmargin` the results in Table 3.1 were obtained.



(a) Loop broken before plant

(b) Loop broken after plant

Figure 3.1: Loop-at-a-time margins: longitudinal axis

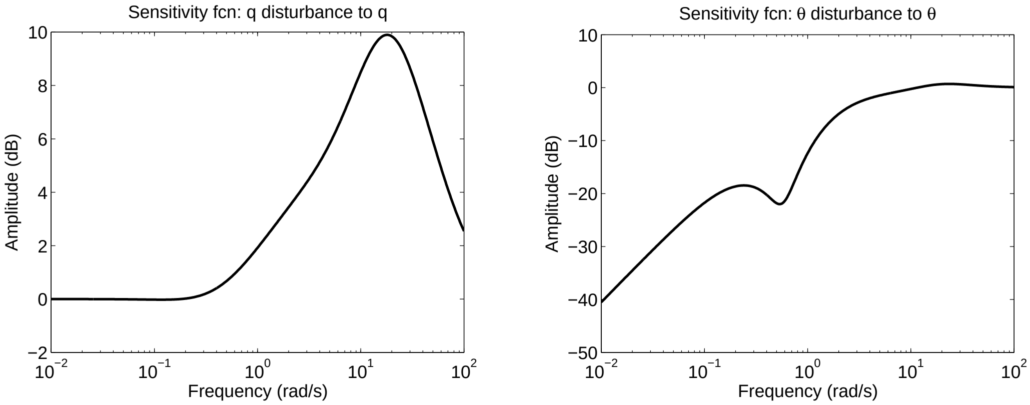
Loop broken...	...after controller	...after plant - q	...after plant - θ
Gain margin	∞	∞	∞
GM frequency $\left(\frac{\text{rad}}{\text{s}}\right)$	∞	0	∞
Phase margin (deg)	114.1006	∞	87.0959
PM frequency $\left(\frac{\text{rad}}{\text{s}}\right)$	4.7141	∞	3.0934
Delay margin (s)	0.4224	∞	0.4914

Table 3.1: Classical margins in longitudinal dynamics

One can conclude that each loop has great margins. Disk margin analysis plots [1] were not generated, since the gain margin is infinite in each loop.

3.1.2 Nominal frequency responses

Unfortunately no accurate information is available about either of the controllers regarding the objectives taken into account at their respective design process. Therefore plausible explanations are given in the sequel why the nominal frequency responses have their correspondigng shape. In case of the longitudinal controller SISO transfer functions θ_{ref} to θ and θ_{ref} to q are considered, since this states are in feedback. Figure 3.7b suggests that the designer aimed to achieve tracking of θ reference signals up to 1 $\frac{\text{rad}}{\text{s}}$. Even though the gain from reference to output θ varies around 1, the significant rolloff starts after the aforementioned frequency. The pitch rate (Figure 3.7a) gain remains less than 1 up to 1 $\frac{\text{rad}}{\text{s}}$, but beyond this frequency it will grow. This implies some criteria on the pitch rate too.


Sensitivity fcn: q disturbance to q

Sensitivity fcn: θ disturbance to θ

(a) Channel q

(b) Channel θ

Figure 3.2: Sensitivity functions in longitudinal axis

Figures 3.2a and 3.2b depict the sensitivity functions for the two feedback channels, respectively. As a consequence of poor tracking, the q channel is not able to reject pitch rate disturbances. Since tracking of θ references was achieved below 1 $\frac{\text{rad}}{\text{s}}$, the effect of disturbances in this channel is significantly smaller.

3.1.3 Robustness analysis

The uncertainties are represented by Δ blocks in this work. But what is included in these? It tries to capture unmodeled dynamics and parametric uncertainties in the system. Effects of unmodeled dynamics are revealed, for instance, in Figure 2.7d. The flight data signal contains extra peaks compared to the model responses. The black curves in Figure 3.3 are Bode magnitude plots of the uncertain transfer function δ_{ele} to θ. The uncertainty was defined as in Figure 3.4, where Δ denotes an `ultidyn` object, and $W = 0.3$. The cyan dashed curves denote plant models, in which the cg position varied by 5% along x axis and 20% along z axis due to for example inaccurate measurement or carrying a payload or installation of a new sensor. In this example, the ultidyn object is able to take into account the effects of the cg variation.

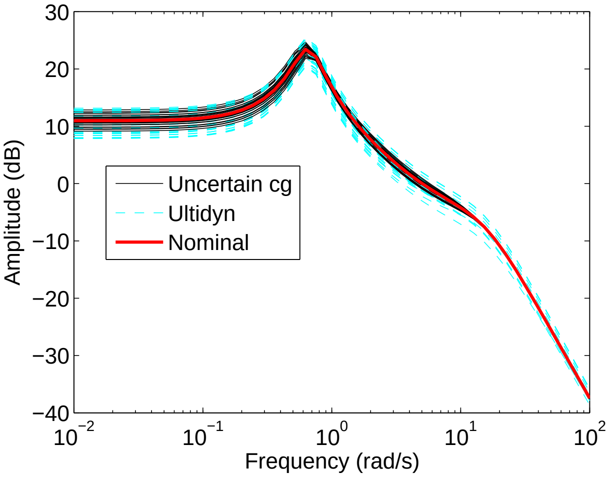


Figure 3.3: Parametric and ultidyn uncertainty on a plant

We focus on the longitudinal axis analysis in the sequel. Multiplicative uncertainty was inserted onto the elevator signal (i.e. the input of the plant). Robust stability, robust performance were computed, as well as worst case gain analysis was peformed on SISO systems as shown in Table 3.2. In each case Δ block represents an uncertain system as defined in A.1. The bound on the perturbation was chosen as $W = 0.3$.

Wind gusts, turbulences cause unwanted motions of the control surfaces while the vehicle is airborne. These effects can be taken into account as disturbances and can be quantified using

Input of the plant	Output of the plant
θ_{ref}	q
θ_{ref}	θ
Elevator disturbance	q
Elevator disturbance	θ

Table 3.2: SISO plants in robustness analysis of longitudinal dynamics

SISO systems with this input. Figures 3.4 and 3.8 show the closed loop block diagram for the two different inputs and output chosen as pitch angle. The systems with pitch rate outputs have similar architecture. The analysis results for the two architecture are presented separately.

Results for reference input to output SISO systems are discussed in the sequel.

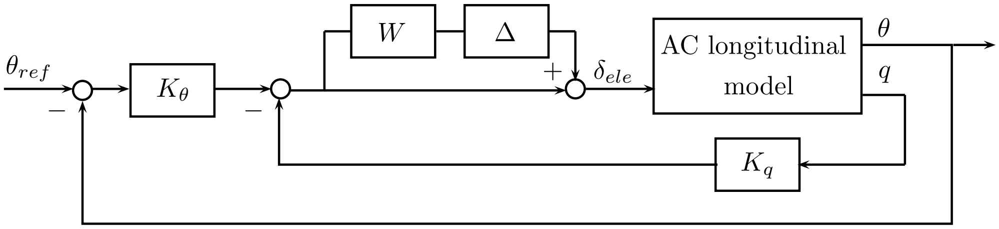


Figure 3.4: Uncertain SISO system: θ reference to θ

Figure 3.5 shows the robust stability μ-plot for the uncertain system in Figure 3.4. We can conclude that the system is robustly stable for the modeled uncertainty. The stability margin has a lower bound 3.3353, upper bound 3.3353. The destabilizing frequency is at 0.01 $\frac{\text{rad}}{\text{s}}$.

The uncertain closed loop system does not achieve robust performance neither with output chosen as q nor chosen as θ. In the first case both the lower and upper bound on the performance margin are 0.2320 at 17.3476 $\frac{\text{rad}}{\text{s}}$. In the second case both the lower and upper bound on the performance margin is 0.9257 at 1.2419 $\frac{\text{rad}}{\text{s}}$ (Figure 3.6).

The uncertain SISO system with output q has a worst case gain lower bound 4.9850, upper bound 4.9864 at 18.1660 $\frac{\text{rad}}{\text{s}}$, which is very close to the peak of the robust performane μ plot. The SISO system with output θ has the worst case gain lower bound 1.0873, upper bound 1.0875 at frequency 1.2534 $\frac{\text{rad}}{\text{s}}$. This frequency, again, almost coincides with the frequency of the peak on the robust performance plot (Figure 3.7).

The robust performance μ plot and the worst case gain plot have an 'unusual' shape, since the nominal closed loop has a Bode amplitude diagram with a peak around 16.5 $\frac{\text{rad}}{\text{s}}$.

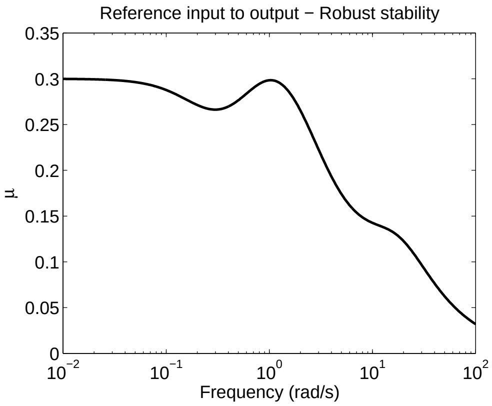
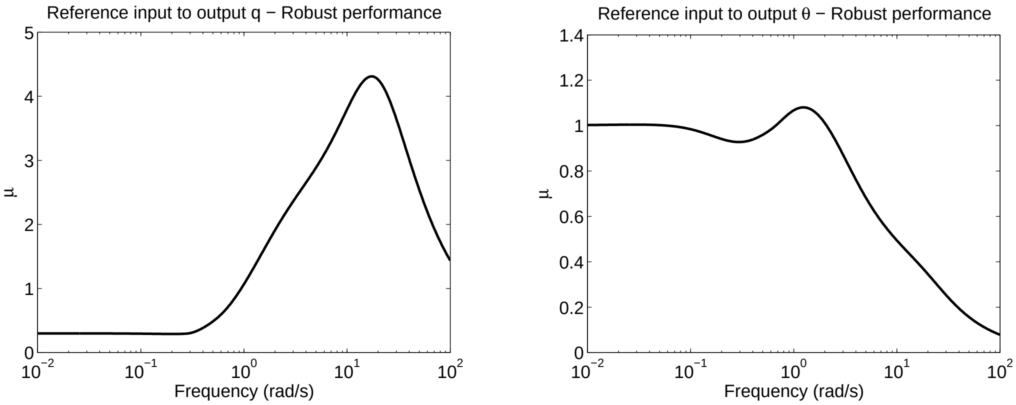


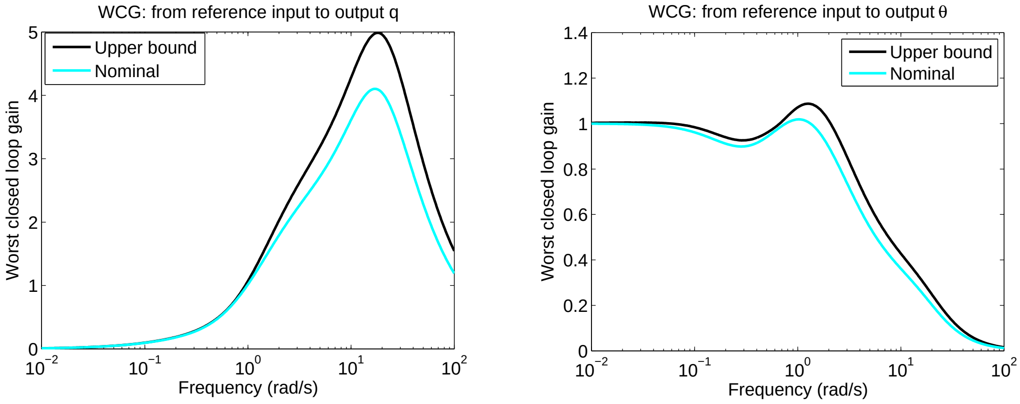
Figure 3.5: Robust stability μ plot - longitudinal axis



(a) q output

(b) θ output

Figure 3.6: Robust performance μ plots - longitudinal axis



(a) q output

(b) θ output

Figure 3.7: Worst case gain plots - longitudinal axis

Results for elevator disturbance input to output SISO systems are discussed in the sequel.

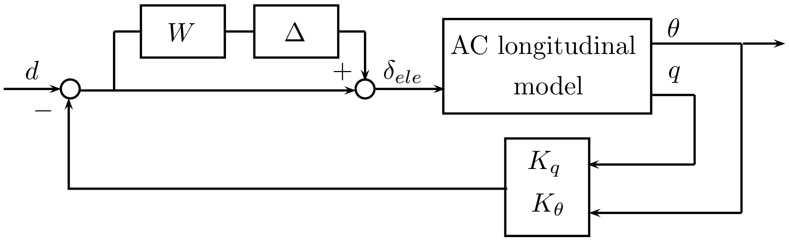
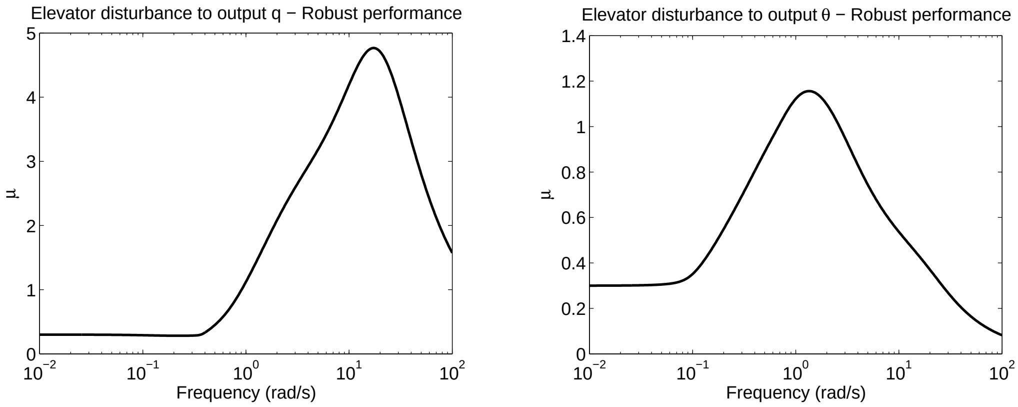


Figure 3.8: Uncertain SISO system: elevator disturbance to θ

The robust stability μ plot is the same as in case of the θ_{ref} input systems.

Robust performance is not achieved in neither of the SISO uncertain systems. The one with ouput q has a robust performance margin 0.2098 (lower and upper bounds are identical) at 17.3476 $\frac{\text{rad}}{\text{s}}$. The closed loop with output θ can be described by a robust performance margin 0.8650 (bounds are the same again) at frequency 1.3494 $\frac{\text{rad}}{\text{s}}$ (Figure 3.9).

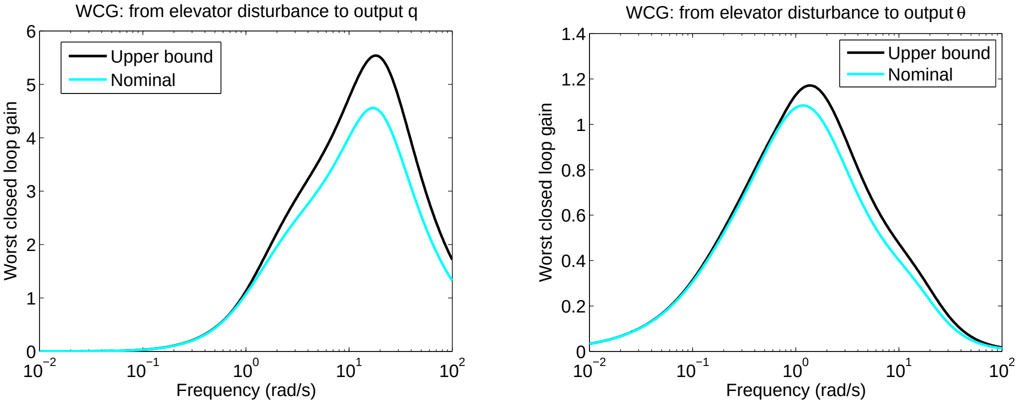
The worst case gains are very close to the robust performance peaks in the frequency range (Figure 3.10). The system with output q has the following bounds on the worst case gain: 5.5379 and 5.5390 at frequency 18.660 $\frac{\text{rad}}{\text{s}}$. With output θ the worst case gain has a lower bound 1.1712 and upper bound 1.1714 at 1.3745 $\frac{\text{rad}}{\text{s}}$.



(a) q output

(b) θ output

Figure 3.9: Robust performance μ plots - longitudinal axis



(a) q output

(b) θ output

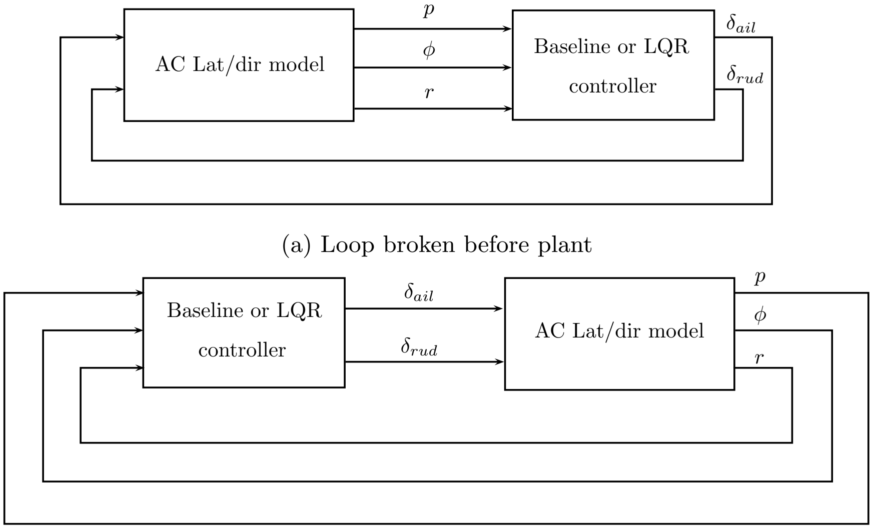
Figure 3.10: Worst case gain plots - longitudinal axis

3.2 Lateral/directional controllers

Baseline and LQR controllers are available to control the lateral/directional axes of the aircraft. In order to have a comprehensive picture about the properties of these controllers, we analyize them paralell.

3.2.1 Classical margins

The classical margins were computed the same way as in case of the longitudinal controller. One loop was broken at a time from the closed loops in Figure 3.11.



(a) Loop broken before plant

(b) Loop broken after plant

Figure 3.11: Loop-at-a-time margins: lateral/directional axes

Classical margins computed in loop in Figure 3.11a can be seen in Table 3.3, while the margins computed in loops broken after the plant (Figure 3.11b) are presented in Table 3.4.

One can conclude that both controller have great classical margins, therefore they can tolerate significant differences in the plant model. There is only one exception, the ϕ loop in Table 3.4 (denoted by *). It does not have gain margin, i.e. when one breaks this loop, the system is unstable. The detailed analysis of this issue can be found in Appendix B. Since this is an extremely slow pole, ($0.00193\,\frac{\text{rad}}{\text{s}}$) there is no need to stabilize it.

Disk margin plots[1] are presented in the sequel. If the ϕ loop is broken, the baseline controller has a non infinite gain margin, thus Figure 3.12 can be drawn. Only in case of output p can

[1] Disk margin plots give information about gain and phase variation at the same time. The x axis indicate the gain variation, while the y axis shows the phase variation at a certain gain variation tick. The curve on the plot denote the limitation in the phase variation for which the system is marginally stable. With gain-phase variations underneath the diagram, the stability of the system is preserved [1].

	δ_{ail}		δ_{rud}	
	Baseline	LQR	Baseline	LQR
Gain margin	∞	∞	∞	∞
GM frequency $\left(\frac{\text{rad}}{\text{s}}\right)$	∞	∞	∞	∞
Phase margin (deg)	106.4048	73.1404	-131.0772	∞
PM frequency $\left(\frac{\text{rad}}{\text{s}}\right)$	7.2058	3.9011	4.0526	∞
Delay margin (s)	0.2577	0.3272	0.9859	∞

Table 3.3: Classical margins in lateral/directional dynamics: loop broken before plant

we make comparison between the two controllers. The disk margin plot confirms the pure data that the LQR controller provides bigger margin than the baseline.

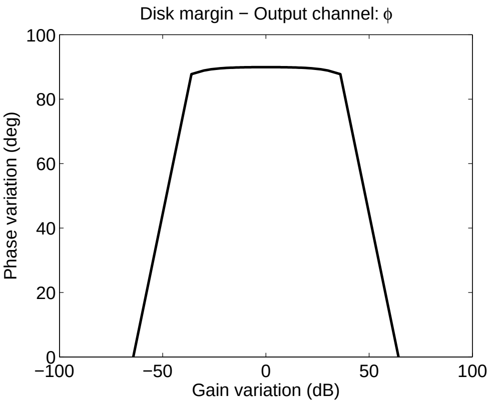


Figure 3.12: Baseline controller disk margin plot - ϕ output

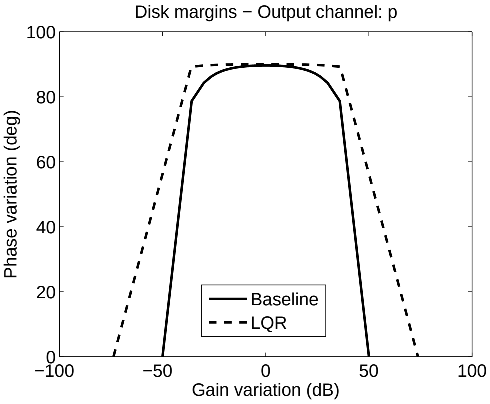


Figure 3.13: Baseline and LQR controller disk margin plot - p output

3.2.2 Nominal frequency responses

Similarly to the longitudinal controller, the exact objective at controller design is unknown. The LQR controller is believed to be the first controller designed for the Ultrastick 25e. The baseline

	p		ϕ		r	
	Baseline	LQR	Baseline	LQR	Baseline	LQR
Gain margin	318.1662	4912	0*	∞	∞	∞
GM frequency $\left(\frac{\text{rad}}{\text{s}}\right)$	0.0943	0.1444	0.0245	∞	0	∞
Phase margin (deg)	∞	∞	83.0768	∞	-131.0772	52.2678
PM frequency $\left(\frac{\text{rad}}{\text{s}}\right)$	∞	∞	3.5670	∞	4.0526	3.4621
Delay margin (s)	∞	∞	0.4065	∞	0.9859	0.2635

Table 3.4: Classical margins in lateral/directional dynamics: loop broken after plant

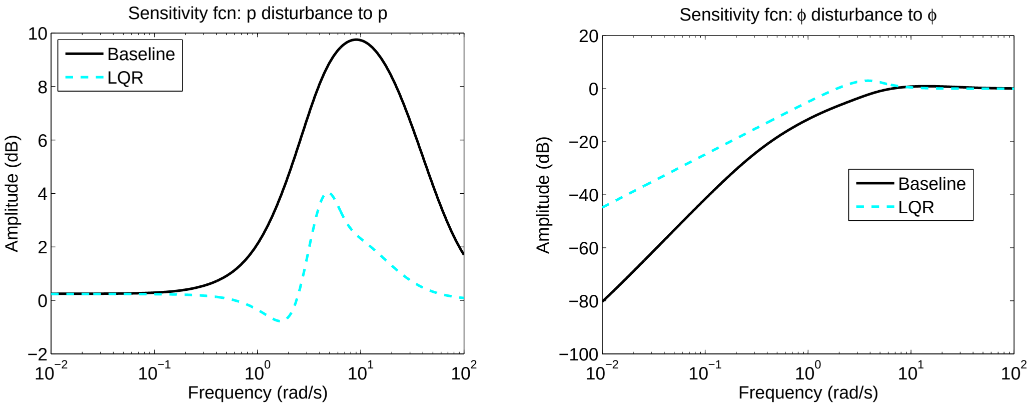
controller is very likely to be an outcome of a semester project in class Design-to-Flight: UAVs.

SISO transfer functions ϕ_{ref} to p and ϕ_{ref} to ϕ are examined, as these will be relevant in the worst case gain analysis. The reference to roll angle transfer functions are depicted in Figure 3.19b for both controllers. One can assume that the objective of the LQR controller was to track bank angle reference signals up to frequency $1\,\frac{\text{rad}}{\text{s}}$. The amplitude plot exposes a gain 1 up to this frequency and attenuates beyond. The objective of the PI design was the same with a high probability, although we can observe some 'overshoot' in the frequency range $0.1-1\,\frac{\text{rad}}{\text{s}}$. Apart from this phenomenon, the baseline controller perfroms well as it has been used in numerous flight tests [11] and has been validated in [3].

Presumably the control design objective contained some requirement on the roll rate. Both the baseline and the LQR design's amplitude plot remains below 1 up to $1\,\frac{\text{rad}}{\text{s}}$ (Figure 3.19a). At higher frequencies, however, this quantity can grow.

The sensitivity functions of channels p and ϕ are shown in Figures 3.14a and 3.14b. The ϕ channel plot confirms that the tracking conditions was set up on the bank angle. The roll rate sensitivity plot shows that disturbances are not rejected in this channel, rather their effect is magnified.

These plots suggest that the baseline design is slightly better in tracking then the LQR. If we examine the nominal frequency responses in Figure 3.19b, it turns out that the LQR controller



(a) Channel p

(b) Channel ϕ

Figure 3.14: Sensitivity functions - lateral/directional axis

provides a better tracking. The interchange in the sensitivity function plots may arise from computation problems. If one calculates the amplitude plot of (T+S), then it will not be strict 0 dB across the entire frequency range. It is below 0 level in frequency ranges $1.05 - 32.5\,\frac{\text{rad}}{\text{s}}$ and $0.7 - 5.28\,\frac{\text{rad}}{\text{s}}$ for the baseline and the LQR contollers, respectively. This implies that the calculation of S is not 100% accurate.

3.2.3 Robustness analysis

Similarly to the longitudinal controller analysis, robust stability, robust performance as well as worst case analysis were performed. Since two controllers are available, comparison of these became possible. Therefore each diagram will contain plots both for the baseline and the LQR controller. Moreover, the lateral/directional dynamics can be controlled via multiple surfaces: ailerons and rudder.

This fact implies that not only effects of diagonal uncertainties can be examined but also full block uncertainties can be included into the loop. These allow cross coupling of the perturbation between the aileron and rudder signals. The weight on each uncertainty was 0.3 both in diagonal and full block setup.

Effects of aileron disturbances are also examined on modified SISO loops.

The described variety of setup results 16 different SISO systems collected in Table C.1. The loops with input ϕ_{ref} are described first.

Since there is no significant difference between the loops, only examples are depicted in the sequel. Figure 3.15 shows the loop with baseline controller, diagonal uncertainty. The input is the bank angle reference, while the output is ϕ (1st row of Table C.1). The combination in row 8 is shown in Figure 3.16, as it has LQR controller (the entire structure of the controller is not shown, substituted by the block K instead), full block uncertainty and an input-output relation ϕ_{ref} to ϕ. A SISO loop is presented in Figure 3.20, where the input is aileron disturbance, output is ϕ, and baseline controller is wrapped around the uncertain plant with full block perturbation (row 15).

Table C.1 also contains the upper and lower bounds on the robust stability margin, robust performance margin and the worst case gains as well as their respective frequencies. It should be noted that the shape of μ plots and worst case gain plots is highly dependent on the nominal frequency response regardless of the controller, uncertainty structure, etc. As we will see the plots associated with ϕ have a fairly constant part at low frequencies and a rolloff at higher regions. Meanwhile the plots associated with output p show a large peak at higher frequencies followed by a rolloff. Tracking of the bank angle can result these characteristics in frequency domain.

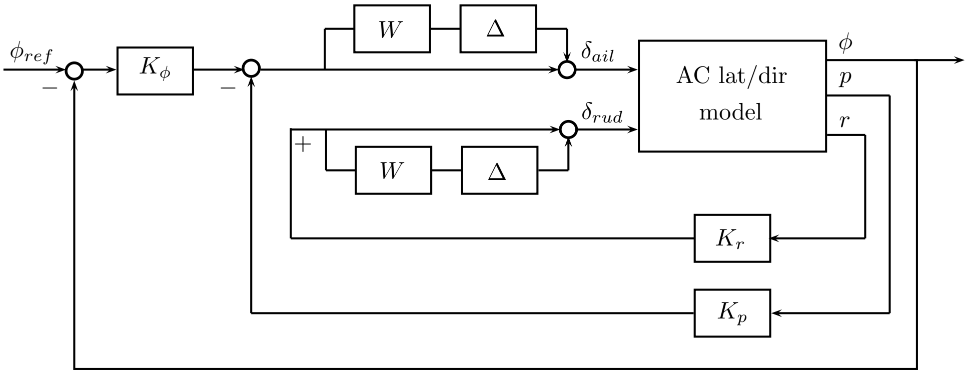


Figure 3.15: SISO model: ϕ reference to ϕ, diagonal uncertainty and baseline controller

Figure 3.17 shows that robust stability is achieved with both controller and with both uncertainty structure. The stability margin of the LQR controller is smaller, when the uncertainty is diagonal. With full block uncertainty, the two peak values are close, but the LQR controller

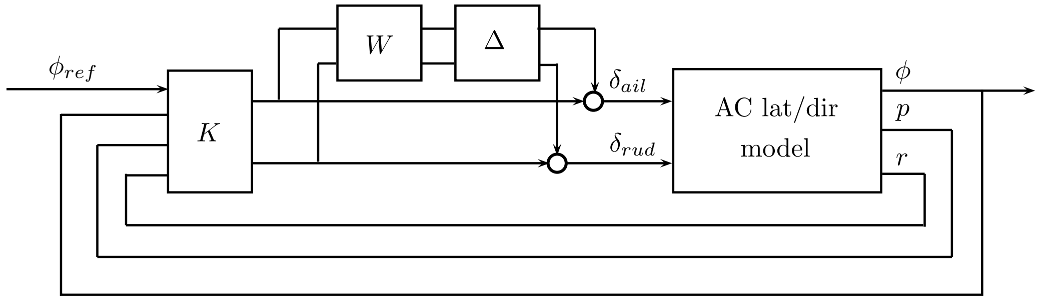


Figure 3.16: SISO model: ϕ reference to ϕ, full block uncertainty and LQR controller

exposes it in a slightly higher frequency.

In robust performance 3.18 there is basically no difference between the diagonal and full block results. Only around the peak values on the ϕ output plots can be observed some discrepancy.

Similarly to the longitudinal axis, the worst case gain plots 3.19 are very similar to the nominal Bode amplitude plots. Again, the diagonal and full block uncertainties result almost the same worst case gain across the frequency range. The ϕ plot (Figure 3.19b) exposes some differences. In the p channel, there is a bigger difference between the nominal and worst case peak frequency responses.

Regarding the ϕ channel the two controllers have similar robust performance margin and worst case gain. This can be the consequence of the same design objective (tracking ϕ reference signals up to $1\,\frac{\text{rad}}{\text{s}}$. The p channel results suggest that in the LQR design process the prevention of large roll rate values has been taken care of. Needless to say that smaller angular rates are favorable from a structural point of view. They will require less control effort from the surface servos, and the structure does not go under quick load changes.

The effects of aileron disturbances on robustness are examined in the sequel. While the vehicle is airborn, wind gusts and turbulences may act directly on the control surfaces. Figure 3.17 shows that robust stability is achieved with both controller and with both uncertainty structure. The stability margin of the LQR controller is smaller, when the uncertainty is diagonal. With full block uncertainty, the two peak values are close, but the LQR controller exposes it in a slightly higher frequency.

In robust performance 3.18 there is basically no difference between the diagonal and full block results. Only around the peak values on the ϕ output plots can be observed some discrepancy.

Similarly to the longitudinal axis, the worst case gain plots 3.19 are very similar to the nominal Bode amplitude plots. Again, the diagonal and full block uncertainties result almost the same worst case gain across the frequency range. The ϕ plot (Figure 3.19b) exposes some differences. In the p channel, there is a bigger difference between the nominal and worst case peak frequency responses.

Regarding the ϕ channel the two controllers have similar robust performance margin and worst case gain. This can be the consequence of the same design objective (tracking ϕ reference signals up to $1\,\frac{\mathrm{rad}}{\mathrm{s}}$. The p channel results suggest that in the LQR design process the prevention of large roll rate values has been taken care of. Needless to say that smaller angular rates are favorable from a structural point of view. They will require less control effort from the surface servos, and the structure does not go under quick load changes.

The numerical values of the peak values of the RS, RP, WCG plots are collected in Table C.1.

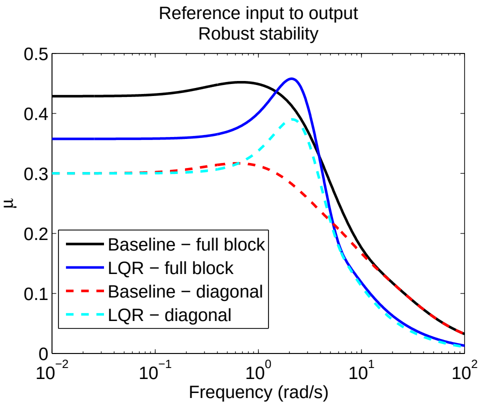
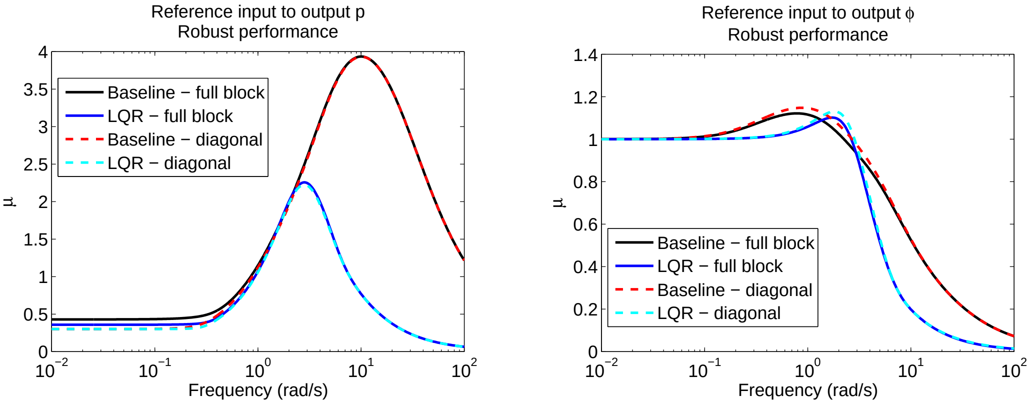


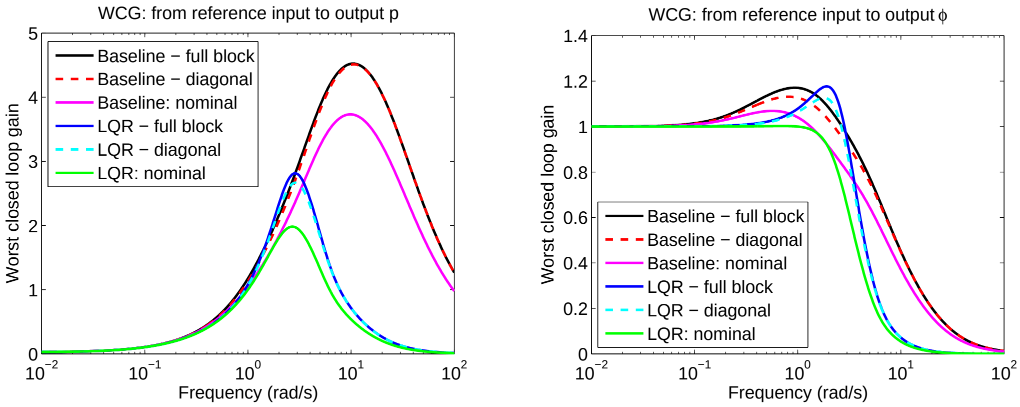
Figure 3.17: Robust stability μ plot - lateral/directional axis



(a) p output

(b) ϕ output

Figure 3.18: Robust performance μ plots - lateral/directional axis



(a) p output

(b) ϕ output

Figure 3.19: Worst case gain plots - lateral/directional axis

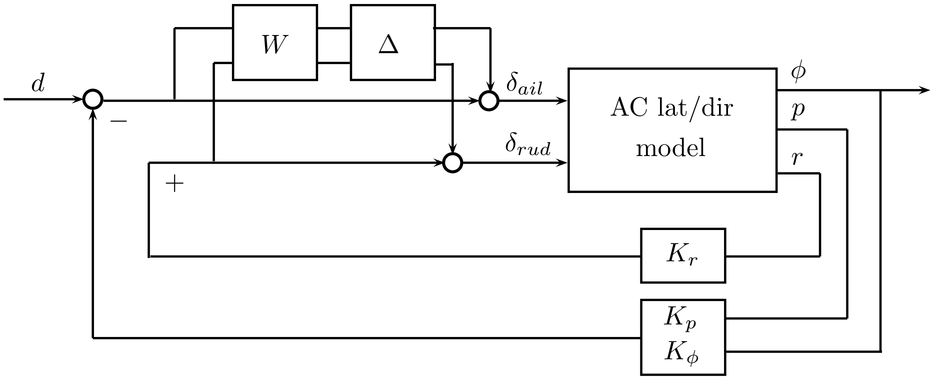
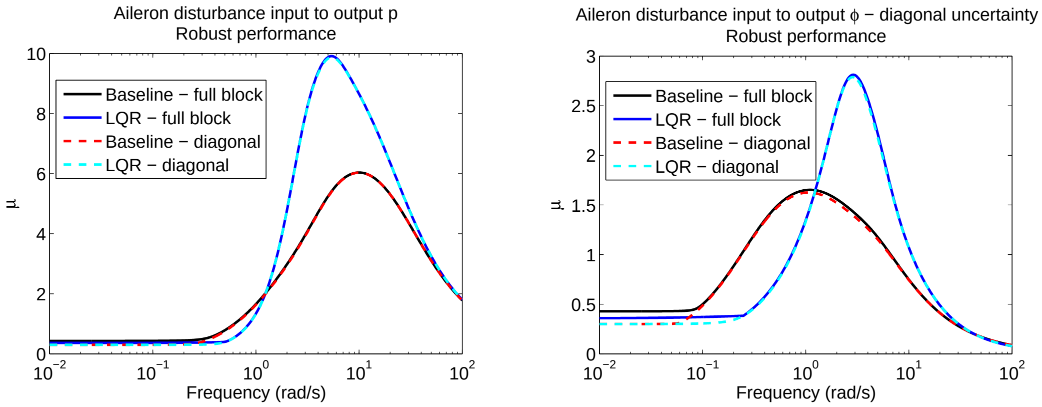


Figure 3.20: SISO model: aileron disturbance to ϕ, full block uncertainty and baseline controller

The effects of aileron disturbances on robustness are examined in the sequel. While the vehicle is airborn, wind gusts and turbulences may act directly on the control surfaces.

The robust stability plot is the same as Figure 3.17.

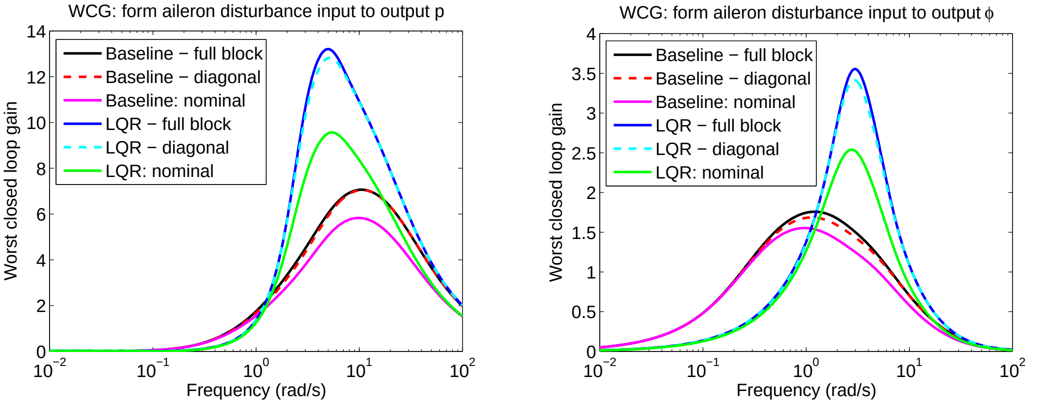
Unlike the SISO systems with ϕ reference, the baseline controller privides significantly smaller worst case gain, when the input is aileron disturbance. The effect of full block uncertainty versus diagonal uncertainty is negligible in all considered SISO loops. The difference between the nominal and perturbed frequency responses are bigger in case of the LQR controller. The numerical values of the peak values of the RS, RP, WCg plots are collected in Table C.2.



(a) p output

(b) ϕ output

Figure 3.21: Robust performance μ plots - lateral/directional axis



(a) p output

(b) ϕ output

Figure 3.22: Worst case gain plots - lateral/directional axis

3.2.4 Worst case models

Not only the worst case gain can be calculated, but also the uncertainty which causes that. In the next step the perturbation can be substituted into the uncertain model, then one obtains an exact model. In this subsection we analyze the worst case uncertainties calculated for SISO loops with ϕ_{ref} as inputs.

In the preceding subsection 30% of uncertainty was included into the nominal linear loop.

These uncertainties were substituted back to the uncertain *nonlinear* system. An aileron reference signal (same as given in Figure 2.7a) was applied to excite the loop. The responses of perturbed system were only slightly different from the nominal nonlinear responses. The only exception was the baseline controller - full block uncertainty combination. Keeping disturbances (wind gusts, turbulence) while airborne in mind one can claim that the mentioned difference is smaller than amplitude of disturbances observed on flight data in the available directory.

If we would like to verify the simulation results with flight test, a greater difference is required between the nominal and worst case responses. These observations implied the next step in which the level of uncertainty was investigated for which the worst case nonlinear loop is at the border of stability. The results are collected in Table 3.5.

Controller	Uncertainty	Level of uncertainty
baseline	diagonal	94 %
baseline	full block	66 %
LQR	diagonal	86 %
LQR	full block	65 %

Table 3.5: Allowable uncertainty levels for robust stability

These levels further support the claim that the baseline controller is more robust than the LQR. Computing worst case uncertainties at different uncertainty level and simulating the nonlinear closed loop allowed to observe the followings. Worst case uncertainties calculated for SISO loops with roll rate output introduced oscillations in the yaw axis with a frequency of 1-2 Hz. The uncertainties coming from SISO loops with ϕ outputs have less significant effect on the response. The fact that the controllers are designed to track bank angle reference signals, can explain these observations.

The worst case roll angle responses differ significantly from the nominal, if the uncertainty level is above 50 %. Another interesting result was that the LQR controller - full block worst case uncertainty combinations did not expose significant oscillations as the amount of uncertainty increased and made the system unstable.

Chapter 4

Nonlinear analysis

4.1 Wcsim tool

Nonlinear systems are usually represented by Simulink models. If one wishes to do a worst case analysis on a nonlinear system, using built-in Matlab tools this is impossible. The command `wcgain` requires a linear uncertain state space model. However, [8, 9] provides a nonlinear optimization based solution: the wcsim package.

It contains a Simulink library with two blocks, both based on the built-in 'To workspace' block: an objective block (RCTObjective) and a constraint block (RCTConstraint). The signals on which the objective is defined, are connected to objective blocks, while the possible constrained signals are connected to constraint blocks. The optimization objective itself is some norm of the signal (L1, L2, L_∞), or simply the pointwise minimum/maximum. The options include user specified objectives written as m files. Either minimum or maximum of the norm is calculated. The listed properties can be set as properties of the Objective block. The Constraint block is set up in a very similar way. The type of contraint is chosen from the same set of norm options as in case of the Objective block. In addition minimum and maximum values on the constrainted signal norm have to be specified.

The nominal block diagram of the system in question is augmented with the Objective and Constraint blocks as well as uncertain state space (uss) model blocks representing the perturbed plant. The uncertainty in the uss block may be defined by uncertain dynamics (`ultidyn`), uncertain real parameter (`ureal`) or uncertain complex parameter (`ucomplex`). The `wcsim` can handle only `ureal` perturbations. The actual calculation is carried out by the Matlab function `wcsim(...)`, invoking an `fmincon` optimization which varies the uncertain parameters. The objective function is defined using the Objective block, the bounds on the optimization are defined by the bounds on the uncertainties, while the possible constraints are specified in Constraints blocks. `fmincon` needs multiple evaluations at each possible perturbation value and their small vicinity along each parameter direction. The system is simulated at each evaluation. The uncertain real parameters are initialized in the middle of their range at the beginning of the computation. Depending on the exact problem the optimization may or may not converge. The outcomes of the optimization are the values of ureal parameters which result in the specified objective (e.g. cause the maximum of the L2 gain of some signal) subject to the possible constraints and are inside the bounds.

4.2 Simulation setup

The nonlinear worst case perturbation analysis was carried out on the model setup shown in Figure 4.1. It contains two complete representation of the closed loop UAV model described in Section 2.2, however the second one is augmented by a perturbation block. The original UAV model has already contained a model reference block for the controllers, therefore it was a straightforward solution to modify the controller block diagrams by adding the perturbations. Thus, the baseline and LQR controllers have perturbed versions as well.

Supporter codes were written to this model which are identical apart from the reference signals. One of them defines ϕ or θ pattern signals, which were designed to excite most of the aircraft dynamics. Figures 2.6a and 2.7a depict these signals. Sinusoidal ϕ and θ inputs in the frequency range $10^{-2} - 10^{2}\,\frac{\text{rad}}{\text{s}}$ are the reference signals in the other one. The amplitude of them is chosen as $0.4\,\frac{\text{rad}}{\text{s}}$ to ensure that the aircraft's attitude angles will not exceed their limitations.

Similarly to the linear worst case gain analysis, a dedicated input-output pair is considered to calculate the gain of SISO systems. The input signal is some kind of θ or a ϕ reference signal as discussed above, while the output signal can be chosen from the set: $[p\ q\ \phi\ \theta]$ by the user. The state choice block in Figure 4.1 represents this. The difference between the chosen signal in the nominal and perturbed systems is calculated. The objective of the minimization process is to maximize the L2 norm of this difference signal. No constraint was taken into account.

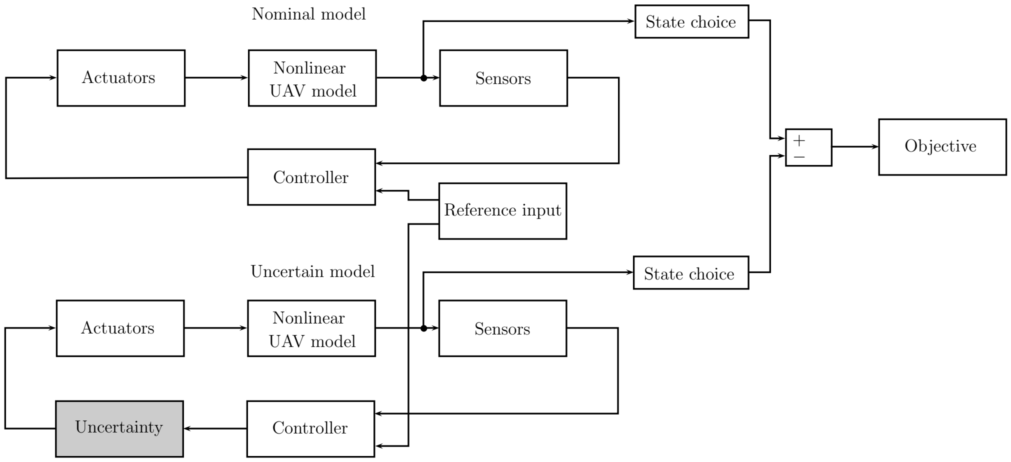


Figure 4.1: Model analysed using wcsim

References [8, 9] note that the convergence of fmincon to the global supremum or infinium is highly dependent on the initial conditions. Several tries were performed with the original wcsim function on linear models, as the results can be validated using the `wcgain` command. At frequencies, where the worst case gain was low (below 1), the optimization converged, however at frequencies of high gains it failed. Significant difference could be observed between wcsim and `wcgain` results. For instance, 4.5 and 3.7 were the computed gains using the two methods, respectively.

One way to enhance the results is to launch multiple optimizations from different initial conditions (at the same frequency), and accept the highest gain from this set as the worst case gain at that particular frequency. The uncertainties are assumed to be uniformly distributed random variables their range. This implies to choose the initial conditions as uniformly distributed random variables from the valid interval. Two possibilities are available regarding the point where the initial conditions are computed. They can be set at the beginning of the simulation, thus the multiple runs at each frequency will start from the same initial conditions. Inaccuarate results may be calculated if none of the initial conditions will lead the optimization to the global supremum. Other approach is to take new set of initial conditions at each frequency. With this method the predestined bad results can be avoided, however there may be significant changes in the worst case gain from frequency to frequency. 10 initial conditions were generated according to the first approach as it does not produce fluctuations in the worst case gain plot.

Since the nonlinear model is 'very linear', it is very likely that perturbations chosen from around their respective bounds will cause the worst case gain. The borders do not necessarily appear among the 10 generated initial condition, therefore the idea above was slightly changed as only 6 random initial conditions were chosen and the lower and upper bounds of the uncertainties are included (with all their possible 4 combinations). With this setup the worst case gain calculation results became acceptable as one can see later in Figure 4.7. Quick calculations proved that the area is very small from where fmincon will converge to the global minimum. If multiple combination of perturbations result the same worst case gain at a freqeuency, the one is selected which arose from the initial condition having a higher index in the array of initial conditions.

A huge increase in the computation time occurs due to the multiple initial conditions. For example generating a worst case gain plot with 25 frequency points lasts 7-8 hours, but with 50 frequency points it can go up to 15 hours (only with one perturbation block, as in the longitudinal axis). The share of running fmincon is about 85%.

The initialization of the model in Figure 4.1 is derived from the SIL simulation setup file. The user can choose a controller (baseline or LQR), form of the perturbation (diagonal in longitudinal axis, diagonal in lateral/directional axis or full block in lateral/directional axis). Further settings involve choosing the state considered as output to the objective block and the type of model (linear or nonlinear).

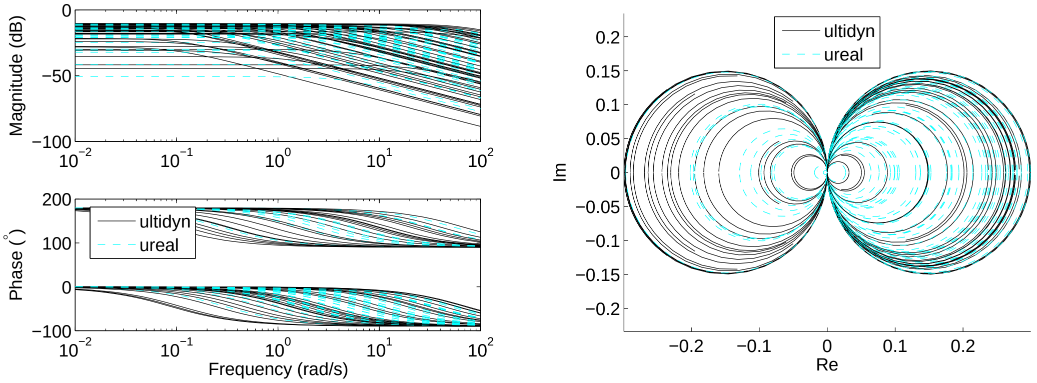
4.3 Uncertainty representation

Following the logic of the linear worst case gain analysis, effects of three different uncertainties were examined. Each perturbation block consists of multiplicative uncertainties. Figure 4.4 shows the perturbation on elevator signal, while Figures 4.5 and 4.6 depict the diagonal and full block uncertainties in lateral/directional axis, respectively.

Compared to the linear analysis, however, two changes had to be made regarding the uncertainty representation. Recall that the uncertainties were defined by first order uncertain linear dynamical systems, using the `ultidyn` Matlab command. Wcsim, however, can handle only real parametric uncertainty, therefore the uncertainty needs to be represented that way. On the other hand the linear analysis was performed on continuous time models (with continuous time uncertainties), while the nonlinear Simulink representation is a discrete time model with a sample time 0.02 s (corresponding to 50 Hz execution of the flight code on the aircraft).

The first order transfer function can be written as $G(s) = \dfrac{A}{Ts+1}$. The two constants are considered as uncertain real parameters. Their range was appointed using Bode and Nyquist plots in the following way. First, continuous time real parameters were calculated by matching frequency domain plots between the continuous and discrete time representations. Figures 4.2a and 4.2b show sample plots. The two kinds of uncertainty representation have fairly matching frequency domain plots. The manual 'tuning' gave the following ranges for the the uncertain parameters:

$$A \in \begin{bmatrix} -0.3 & 0.3 \end{bmatrix}\ , \text{ with a nominal value: } A_n = 0.1778\,,$$
$$T \in \begin{bmatrix} 0.01 & 10 \end{bmatrix}\ , \text{ with a nominal value: } T_n = 0.1\,.$$



(a) Bode plot

(b) Nyquist plot

Figure 4.2: Frequency domain plots of perturbations built using ultidyn and ureal commands

Numerous discretization methods are discussed in the literature [6]. In this case the variable s was approximated using the Tustin method, which uses the following substitution:

$$s = \frac{2}{T_s}\frac{z-1}{z+1}, \text{where } T_s \text{ denotes the sample time.}$$

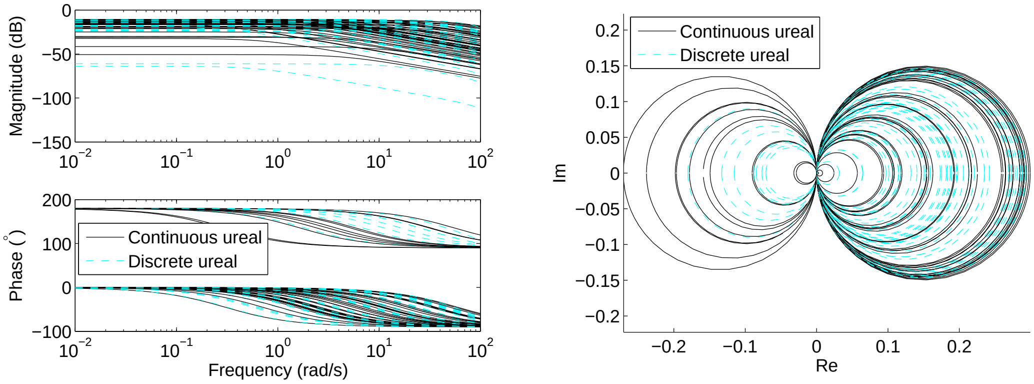
Applying the same notation as before the first order transfer function of is given by:

$$G(z) = \frac{(A \cdot T_s)z + A \cdot T_s}{(2T + T_s)z + (T_s - 2T)} = \frac{\dfrac{A \cdot T_s}{T_s - 2T} z + \dfrac{A \cdot T_s}{T_s - 2T}}{\dfrac{2T + T_s}{T_s - 2T} z + 1} = \frac{A_d}{T_d \cdot z + 1}. \tag{4.1}$$

Other discretization methods were also tested, but the results were less accurate at high frequency. Since the Nyquist frequency $\left(0.5 \cdot 314.15 = 157.08 \, \frac{\text{rad}}{\text{s}}\right)$ is close to the maximum considered frequency $\left(100 \, \frac{\text{rad}}{\text{s}}\right)$, there was a significant discrepancy between the continuous and discrete time responses at that frequency range, especially in the phase.

In the Matlab realization, new uncertain parameters were not introduced, i.e. A_d and T_d are only transformations of the original ureal parameters A and T. This implies that in the results in the sequel values for A and T will be given, but one has to take into account that in fact the nominal model is perturbed by (4.1).

Figures 4.3a and 4.3b show samples of continuous and discrete time ureal uncertainty representations.



(a) Bode plot

(b) Nyquist plot

Figure 4.3: Frequency domain plots of perturbations built using ureal command

Figure 4.4, 4.5 and 4.6 represent the Simulink realizations of the uncertainty blocks in the longitudinal and lateral/directional axes, respectively. The transfer function blocks are identical, therefore in the analysis only one T and one A value is obtained in each run. They contain the discrete time transfer function given by (4.1).

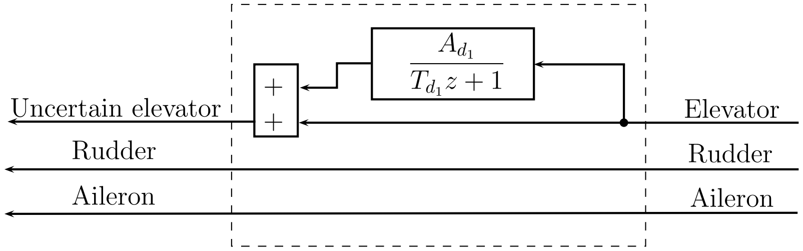


Figure 4.4: Diagonal uncertainty in longitudinal axis

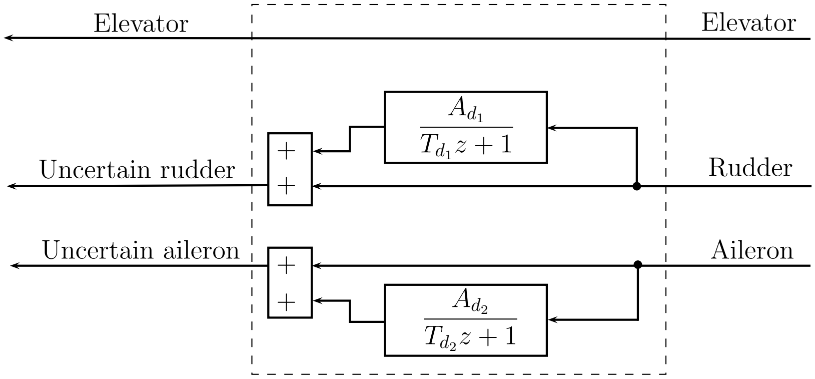


Figure 4.5: Diagonal uncertainty in lateral/directional axis

4.4 Validation of the wcsim code package

The aforementioned possibility to choose linear model in the supporting code is introduced in order to validate certain code snippets. More exactly to test whether the main loop calculating the worst case gain, as well as the modified wcsim function work as expected. Figure 4.7 demonstrates that the wcsim code package performs well. The roll rate output was chosen as the worst case gain is high in this case. Even though there is a sightful discrepancy at higher frequencies, the biggest gains are calculated fairly correcly. This test was performed for each uncertainty-controller case.

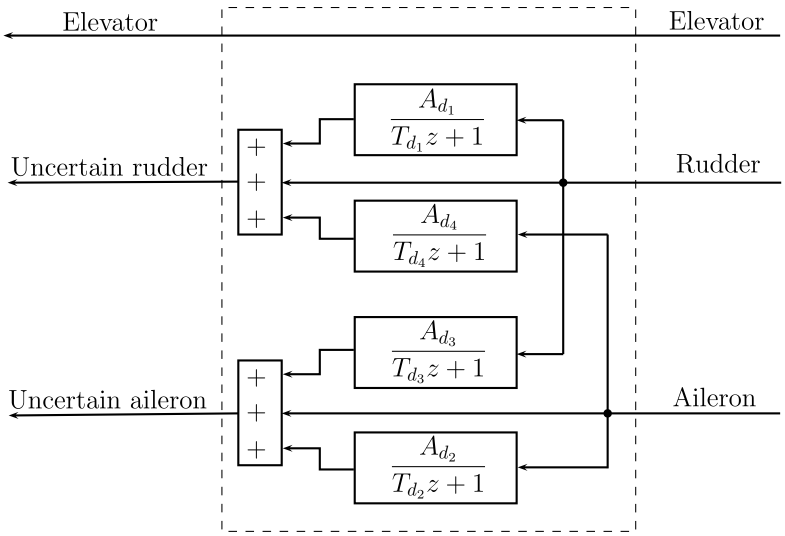

Elevator
Elevator
Uncertain rudder
Rudder
Uncertain aileron
Aileron
+
+
+
+
+
+
$\frac{A_{d_1}}{T_{d_1}z+1}$
$\frac{A_{d_4}}{T_{d_4}z+1}$
$\frac{A_{d_3}}{T_{d_3}z+1}$
$\frac{A_{d_2}}{T_{d_2}z+1}$

Figure 4.6: Full block uncertainty in lateral/directional axis

4.5 Nonlinear worst case perturbations

4.5.1 Angle pattern inputs

Both linear and nonlinear systems were included in the analysis, as `wcgain` can only handle sinusoidal inputs. Both baseline and LQR controllers were examined. The systems were perturbed by elevator, diagonal aileron/rudder and full block aileron/rudder uncertainties.

The numerical results are collected in Table D.1. In most of the cases variable A had its highest possible value, as it represents the low frequency gain of the first order transfer function. The reference signal has no high frequency component, thus the the higher the low frequency gain of the perturbation is the higher worst case gain is achieved. Variable T sticked to the lower boundary value in many case, however it varies in the allowed interval. The lower the time constant is, the higher is the frequency, where the Bode amplitude plots starts attenuating. Same explanation is valid as in case of variable A.

Similar computations were performed on the linear model. The results are very similar to the nonlinear ones. Table C.3 collects the numerical results.

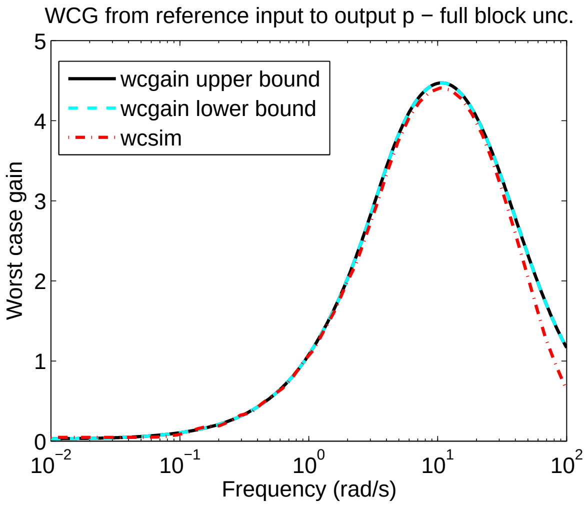


Figure 4.7: Wcsim code validation

4.5.2 Sinusoidal inputs

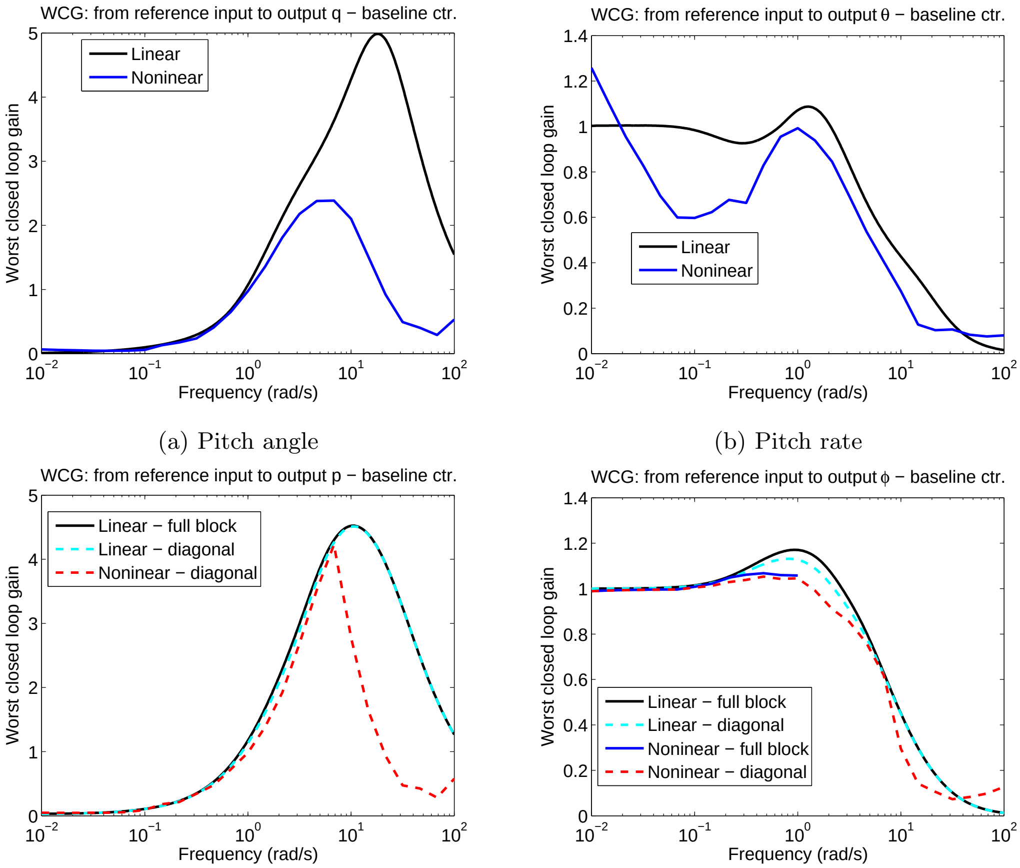
The same set of models were tested as with angle pattern reference inputs. Table D.2 constains the numerical results. The maximal possible A is dominant in these cases as well, and T has a greater variation again. The obtained worst case gain plots are shown in the sequel.

The longitudinal axis the difference between the linear and nonlinear plots is clearly visible. With pitch rate output, the peak of the worst closed loop gain plot is significantly smaller, and achieved at a lower frequency (Figure 4.8a). The low frequency gains in Figure 4.8b are very different compared to the linear case.

In the lateral/directional axis baseline controller and LQR controller results are depicted in different set of Figures: 4.8c, 4.8d and Figure 4.9, respectively. In case of the baseline controller, the SISO system with roll rate has a much sharper, and lower peak than the linear results. This may be caused by a not enough fine frequency space in the computing. With output ϕ and full block uncertainty there is a particular problem with the simulation, therefore only the low frequency half of the diagram is plotted. Multiple launches of the simulation failed at the same frequency point. The Atmosphere model gives NaNs at this point. This never happened with any other SISO system/perturbation combination. The issue is under further investigation. With diagonal uncertainty the diagram follows the linear ones.

The nonlinear plots with LQR controller and output p are the closest to the linear diagrams. The systems with ϕ output have smaller low frequency worst case gains, than the corrsponding linear systems.

In general one can appoint that as the number of uncertainties increase, the more local maximums are introduced in the 'gain space', and fmincon less likely finds the global one. The small scale differences between uncertainties (for example order of magnitude 0.1) cannot be found using this setup. The calculations should be carried out at a finer frequency grid (for example 50 points).



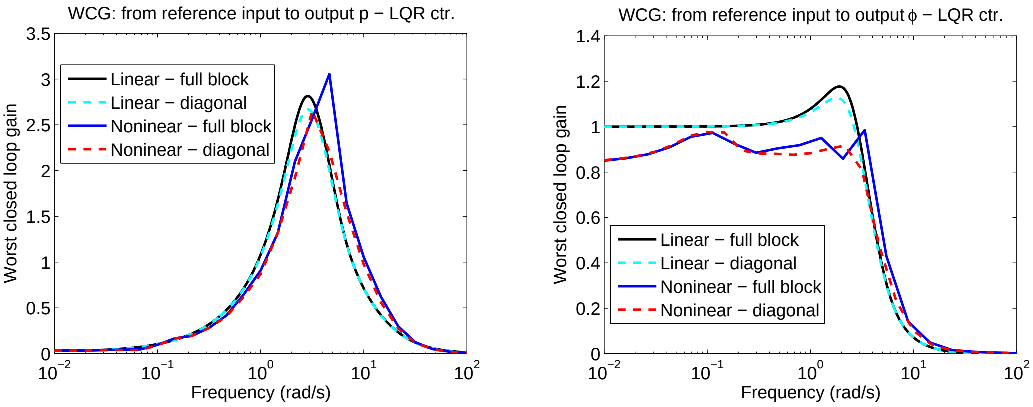
(a) Pitch angle

(b) Pitch rate

(c) Roll rate

(d) Roll angle

Figure 4.8: Nonlinear worst case gain plots with baseline controller



(a) Roll angle

(b) Roll rate

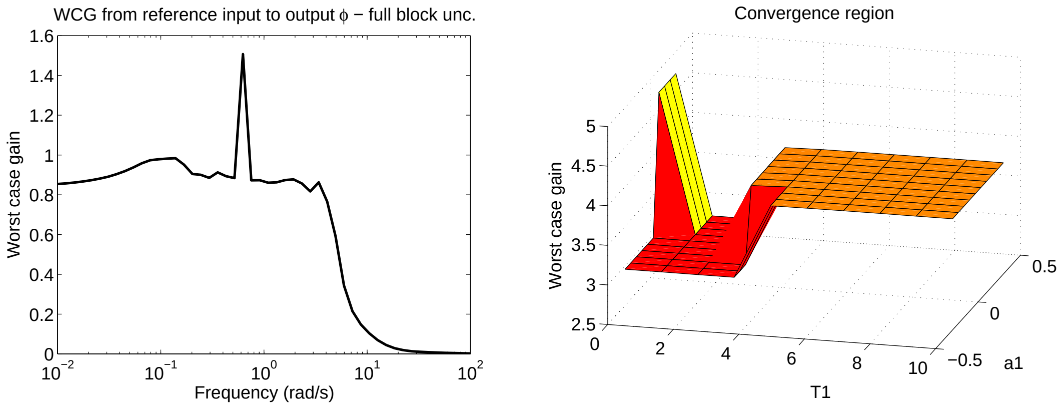
Figure 4.9: Nonlinear worst case gain plots with LQR controller

4.6 Drawbacks of the method

One clear drawback of this solution has already been mentioned: the enormous running time.

Nevertheless in some cases fmincon is not able to find a feasible solution, therefore its run is terminated. This phenomenon may occur at multiple frequencies. Figure 4.10a at the middle frequency region terminations happened which is proved by the lack of the smoothness. The optimization variables (A and T) may be out of their bound causing a huge peak in the worst case gain plot at a certain frequency. in this example the LQR controller on the nonlinear plant was perturbed by full block aileron/rudder disturbances. At frequency $0.6251 \frac{\text{rad}}{\text{s}}$ the problem described above occured. The worst case uncertainties indicated by the code are $A = 0.3178$, $T = 0.0099$, where T is out of bound, therefore it is clearly not the right solution.

Figure 4.10b represents the calculated worst case gain with different initial conditions in case of a calculation on the linear model, when q was the chosen output. The worst perturbation is $T_1 = 0.01$ as well as $a_1 = 0.3$. Only the solution and other initial condition in its close vicinity led the calculation to the solution resulting the globally biggest gain. With other initial conditions fmincon found only a local maximum. This issue becomes more significant, as the number of uncertain parameters increase. The more parameters introduce more local maximums. Due to this phenomenon, the results showing worst case gain in the lateral/directional dynamics are less reliable.



(a) Wcsim termination

(b) Wcsim convergence region

Figure 4.10: Wcsim drawbacks

4.7 Validation via flight tests

Several flight tests were conducted to validate the SIL simulation results. They took place in UMore Park, Rosemount on the 5th of November. Only low speed (1-2 mph) wind blew that morning, therefore the data acquired lacks of big oscillations due to disturbances.

Six experiments were designed with different input signals, controllers and uncertainty types. A compiled flight software has been prepared for each experiment, thus the pilot had to land after each experiment to change the the flight software and acquire data. The pilot flew the aircraft in an oval shape trajectory over the airfield, and engaged the autopilot software only in upwind direction. 3-5 repetitions were done to acquire sufficient amount of data. The length of engagements varied in time depending on how comfortable the pilot was with the motion of the aircraft.

In general, we expect the biggest differences between the nominal and perturbed responses in the yaw axis, as this part of the dynamics is modeled the least accurately. Moreover, the control authority in the yaw axis is limited, since only a yaw damper is available. These two properties make the yaw axis the most sensitive.

A table collects the settings made at each experiment in the following subsections. The obtained plots can be found in Appendix E. Attitude angles and rates, as well as surface commands were measured.

Unfortunately the flight test were carried out on wrong calculations, i.e. the perturbations had the following form:

$$\Delta_{diag} = \begin{bmatrix} \delta_1 & 0 \\ 0 & \delta_1 \end{bmatrix}, \quad \Delta_{full} = \begin{bmatrix} \delta_1 & \delta_1 \\ \delta_1 & \delta_1 \end{bmatrix}.$$

as diagonal and full block uncertainties, respectively. This solution is wrong, because it gives a more conservative solution, as wcgain/wcsim can vary only two parameters: A and T. Each entry of the perturbation matrices is supposed to be independent from the others.

The method of the validation process, however, was correct.

4.7.1 Experiment 1

Analysis tool	wcgain
Model	linear
Controller	baseline
Uncertainty	full block
Input signal	$\sin(1.016 \cdot t)$
Output	ϕ
Fault	none
Worst case perturbation	$\begin{bmatrix} \frac{s^3 - 2.9721s^2 + 2.9432s - 0.9719}{s^3 - 2.9621s^2 + 2.9249s - 0.9628} & \frac{s^3 - 0.0091s^2 + 0.0183s - 0.0092}{s^3 - 2.9621s^2 + 2.9249s - 0.9628} \\ \frac{s^3 - 0.0091s^2 + 0.0183s - 0.0092}{s^3 - 2.9621s^2 + 2.9249s - 0.9628} & \frac{s^3 - 2.9721s^2 + 2.9432s - 0.9719}{s^3 - 2.9621s^2 + 2.9249s - 0.9628} \end{bmatrix}$

Note that the worst case perturbation is given with 3 digits after the decimal point. It is essential to use more accurate descriptions of the perturbations in the flight code, since the absolute value of the poles of these transfer functions is very close to 1. The 3 digits representations may have unstable poles causing the entire closed loop unstable.

The SIL testing promised a significant difference in the yaw axis in this experiment.

With this uncertainty the roll axis is supposed to give a fairly matching response with the nominal system. In the pitch axis responses, however, greater differences appear between the nominal and perturbed responses. As it is expected, the greatest differences can be measured in the yaw axis, since only the yaw damper has authority to control this part of the dynamics. The surface signals reflect the same behaviour.

4.7.2 Experiment 2

Analysis tool	wcsim
Model	linear
Controller	LQR
Uncertainty	full block
Input signal	$\sin(2.81 \cdot t)$
Output	p
Fault	none
Worst case perturbation	A=0.3178, T=1

The outcomes of this experiment have been expected similar to the previous one, nevertheless the LQR controller was in the loop.

The LQR controller does not change the 'sensitivity' of the parts of the dynamics. Therefore the roll axis responses will the closest to the nominal, the pitch and yaw follow that.

4.7.3 Experiment 3

This perturbation introduces high frequency oscillations in the roll/yaw axis surfaces when baseline controller was used. The vehicle did not react to these oscillations, as it can be seen on Figures E.5. The frequency of these oscillations is roughly $157\,\frac{\mathrm{rad}}{\mathrm{s}}$. This value is much higher than the actuator rate limit $2.62\,\frac{\mathrm{rad}}{\mathrm{s}}$, thus it cannot carry out the high frequency oscillations.

The aileron deflection command has a bigger amplitude than the rudder signal, thus oscillations with the same amplitude would cause smaller discrepancy from the nominal signal. The results show, however, that the rudder signal goes under higher amplitude high frequency oscillations.

Analysis tool	wcsim
Model	nonlinear
Controller	baseline
Uncertainty	full block
Input signal	$\sin(6.81 \cdot t)$
Output	p
Fault	none
Worst case perturbation	A=0.3178, T=0.01

4.7.4 Experiment 4

Analysis tool	wcsim
Model	linear
Controller	baseline
Uncertainty	full block
Input signal	ϕ pattern
Output	p
Fault	none
Worst case perturbation	A=0.3178, T=1

5 repetitions have been performed, however only one is shown in the figures to keep them legible. The perturbation is the same as in experiment 3, however it caused significant deficiencies in the ϕ response. It introduced extra peaks in the roll rate. The pitch and yaw responses follow only the trend of the nominal response, otherwise oscillations are introduced. The high frequency oscillations in the surface signals are, again, in the $10^2 - 10^3$ frequency region, which cannot be realized by the actuator.

4.7.5 Experiment 5

Analysis tool	wcsim
Model	nonlinear
Controller	baseline
Uncertainty	diagonal
Input signal	$\sin(0.625 \cdot t)$
Output	p
Fault	none
Worst case perturbation	A=0.0078, T=1.3877

This experiment was introduced to show an example, where the worst case perturbation does not cause significant difference in the responses compared to the nominal ones in the SIL tests. The flight test results confirm the previous statement.

4.7.6 Experiment 6

Experiment 6 is essentially experiment 1, but an elevator surface fault was injected in order to investigate the effects of the worst case perturbation and the fault together. The ramp fault started at the 7th second after the autopilot was engaged. In 13 second it increased from 0° to 10° One can conclude that the results are not significantly worse than in the first experiment. This means that the controller was able to handle these particular ’disturbances’. We cannot conclude anything in general about faulted worst case responses, as only one test was carried out with fault.

Analysis tool	wcgain
Model	linear
Controller	baseline
Uncertainty	full block
Input signal	$\sin(1.016 \cdot t)$
Output	ϕ
Fault	ramp
Worst case perturbation	$\begin{bmatrix} \frac{s^3 - 2.9721s^2 + 2.9432s - 0.9719}{s^3 - 2.9621s^2 + 2.9249s - 0.9628} & \frac{s^3 - 0.0091s^2 + 0.0183s - 0.0092}{s^3 - 2.9621s^2 + 2.9249s - 0.9628} \\ \frac{s^3 - 0.0091s^2 + 0.0183s - 0.0092}{s^3 - 2.9621s^2 + 2.9249s - 0.9628} & \frac{s^3 - 2.9721s^2 + 2.9432s - 0.9719}{s^3 - 2.9621s^2 + 2.9249s - 0.9628} \end{bmatrix}$

Chapter 5

Conclusion

We examined robustness properties of a pair of controllers available for a fixed-wing aircraft. In the first part of the thesis linearized models of the vehicle were considered. Classical margins analysis has shown that both controllers have excellent gain, phase and time domain margins. Assuming 30% uncertainty on the system, robust stability is achieved, while robust performance is not. According to the worst case gain analysis, the perturbations do not change the shape of the frequency response, but they increase the gain, especially at the nominal peak value.

Worst case gain analysis of the nonlinear models was possible using a software package called wcsim. Despite several shortcomings of this solution, worst case gain plots of the nonlinear system have been generated, which are very close to the linear results in most of the cases.

Flight experiments were set up with different uncertainties. Unfortunately these uncertainties were results of a more coservative analysis. Despite this fact, they showed that perturbed system simulations result almost the same response signals as the real flight test vehicle.

As the nonlinear system is very linear, thus the linear results approximate the noninear ones.

Several directions can be identified for further development. For example very nonlinear guidance trajectories may have effects on the results. Furthermore, the fmincon function can be replaced in the wcsim tool. Mathworks is currently developing a Global Optimization toolbox. The application of these tools may result shorter running times in the nonlinear analysis.

Bibliography

[1] Matlab `dmplot` command description. `http://www.mathworks.com/help/robust/ref/dmplot.html`. Retreived: 06/19/2013.

[2] A. Dorobantu. *Test Platforms for Model-Based Flight Research.* PhD thesis, University of Minnesota, 2013.

[3] A. Dorobantu, W. Johnson, F. A. Lie, B. Taylor, A. Murch, Y. C. Paw, D. Gebre-Egziabher, and G. Balas. An airborne experimental test platform: From theory to flight. In *Proceedings of the American Control Conference*, 2013.

[4] A. Dorobantu, A. Murch, B. Mettler, and G. Balas. Frequency domain system identification for a small, low-cost, fixed-wing UAV. In *AIAA Guidance, Navigation and Control Conference*, 2011.

[5] A. Dorobantu, A. Murch, B. Mettler, and G. Balas. System identification for small, low-cost, fixed-wing unmanned aircraft. *Journal of Aircraft*, 50(4):1117–1130, 2013.

[6] G. F. Franklin, J. D. Powell, and M. L. Workman. *Digital Control of Dynamic Systems.* Addison-Wesley, 3rd edition, 1997.

[7] Y. C. Paw. *Synthesis and Validation of Flight Control for UAV.* PhD thesis, University of Minnesota, 2009.

[8] P. J. Seiler, G. J. Balas, and A. Packard. Worst-case simulation with the GTM Design Model. In *ESA-CNES-DLR Workshop on Worst Case Analysis Tools for Guidance Navigation & Control Systems*, 2012.

[9] P. J. Seiler, G. J. Balas, and A. Packard. Worst-case simulation in Matlab/Simulink, May 2013.

[10] S. Skogestad and I. Postlethwaite. *Multivariable feedback control*. John Wiley & Sons, Second edition, 2005.

[11] B. Taylor, A. Dorobantu, and G. Balas. University of Minnesota UAV Research Group. `http://www.uav.aem.umn.edu`.

[12] K. Zhou, J. C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice Hall, 1995.

Appendix A

Uncertain systems, robustness analysis

A.1 Uncertainty modeling

In most cases the mathematical model of a system does not capture the real system's dynamics entirely. There may be multiple reasons for that, for example some states cannot be measured, handling a high fidelity model would require unreasonable computation effort, analysis and design tools are not available for certain nonlinear systems, etc. Therefore the behavior of the model slightly differs from the real system's response.

With uncertainties, however, we can model the aforementioned differences. Multiplicative and additive uncertainty structures are the most common [10]. Figure A.1 shows a block diagram of multiplicative uncertainty structure. P represents the mathematical model of the plant, referred as nominal system. Δ denotes the uncertainty taken from the set $\boldsymbol{\Delta}$:

$$\boldsymbol{\Delta} = \{\Delta(s) : \quad ||\Delta(s)||_\infty < 1\}\,. \tag{A.1}$$

W is a weighting function to be chosen by the designer. The nominal plant together with the uncertainty model build up the uncertain plant, denoted by $\hat{P}$.

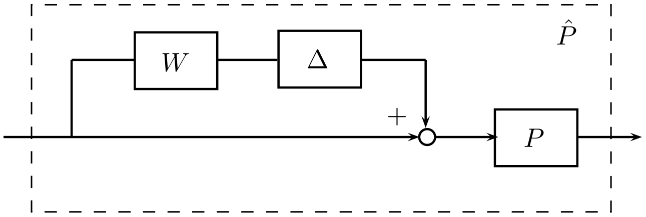


Figure A.1: Multiplicative uncertainty structure

Figure A.2 shows an example of an uncertain closed loop with a multiplicative uncertainty structure.

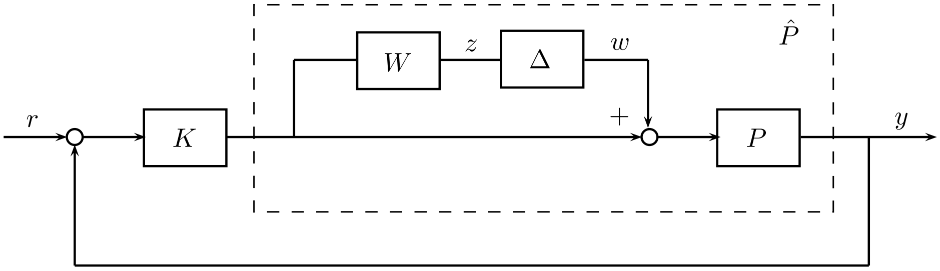


Figure A.2: Uncertain loop

This architecture can be brought into the form in Figure A.3 using block diagram manipulations. Here, the block Δ may denote more than one perturbation, i.e. all the uncertain parts of the loop are grouped into one hypermatrix. The block N, in turn, refers to everything else in the loop such as plant model, controller parts, weighting functions.

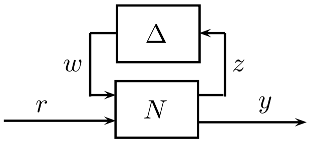


Figure A.3: The N - Δ structure

Other ways (such as parametric uncertainties) are also used to capture variations in the models [10].

Matlab handles the uncertain models in a different way from the exact models. They are stored as set of models, which sets can be randomly sampled resulting exact models. Another way to make the uncertain model certain is substituting in some sort of given uncertainty model (which is usually the worst case uncertainty, defined below).

A.2 Robustness

Let us define set of models $\mathcal{M}$ as:

$$\begin{aligned} \mathcal{M} = \{ P(1+W\Delta) : \quad & |\Delta(\mathrm{j}\omega)| < 1\,,\ \forall\omega \\ \#\mathrm{RHPP}(P) &= \#\mathrm{RHPP}[P(1+W\Delta)]\}\ , \end{aligned} \tag{A.2}$$

where RHPP is a shorthand notation on right half plane pole.

$$\tag{A.3}$$

Figure A.2 depicts the system architecture.

Conditions can be established to ensure that this system remains stable for any uncertainty from the $\mathbf{\Delta}$ set. The following theorem introduces the robust stability condition.

Theorem A.2.1 (Robust stability)**.** *Assume that the nominal loop is stable, i.e. K stabilizes P. Define $T = \dfrac{L}{1+L}$, where $L = PK$. K stabilizes all $\hat{P} \in \mathcal{M}$ if and only if $||WT||_\infty \leq 1$.*

Proof is detalied in [10]. Introduce M as $M = N_{11}$, i.e. the transfer function from y to z.

Notion of robust stability margin can be formalized as:

Definition A.2.1 (Robust stability margin)**.** *The robust stability margin K_M is defined as follows:*

$$K_M = \min_{\Delta \in \mathbf{\Delta}} ||\Delta||\,, \tag{A.4}$$

such that the closed loop in Figure A.3 goes unstable.

If $K_M > 1$, then the system is stable for all uncertainties in $\mathbf{\Delta}$. If $K_M < 1$, then there are perturbations in set $\mathbf{\Delta}$, which destabilize the uncertain closed loop. (The subscript 'M' refers to the generalized plant block in Figure A.3.

In the sequel, we introduce the notion of structural singular value (denoted by μ).

Definition A.2.2 (Structural singular value)**.** *The structural singular value (μ) can be computed as:*

$$\mu(M) = \begin{cases} 0\,, & \textit{if } \Delta \in \mathbf{\Delta} \textit{ causes instability,} \\ \dfrac{1}{\min \bar{\sigma}(\Delta) \quad | \det(I - M\Delta) = 0}, & \textit{otherwise.} \end{cases} \tag{A.5}$$

Only lower and upper bounds are given for the robust stability margin, since computing the exact value would be difficult. It has been proven that the exact value lies between these bounds.

The μ plots indicate the inverse of the stability margin at each frequency ω. Moreover the marginally stable uncertain system has poles at $\pm \mathrm{j}\omega$.

The notion of robust performanceis introduced in the sequel. Robust performance is achieved, if the performance objective is satisfied for any perturbation in the allowed set $\mathbf{\Delta}$. Let us define the transfer function F between the exogenous input r and the exogenous output y as $F = N_{22} + N_{21}\Delta(I - N_{11}\Delta)^{-1}N_{12}$. The robust performance sets up a condition on this transfer function as:

Definition A.2.3 (Robust performance)**.** *If the system in Figure A.3 is robustly stable, then it may achieve robust performance if*

$$||F||_\infty < 1, \quad \forall \Delta \in \mathbf{\Delta}\,. \tag{A.6}$$

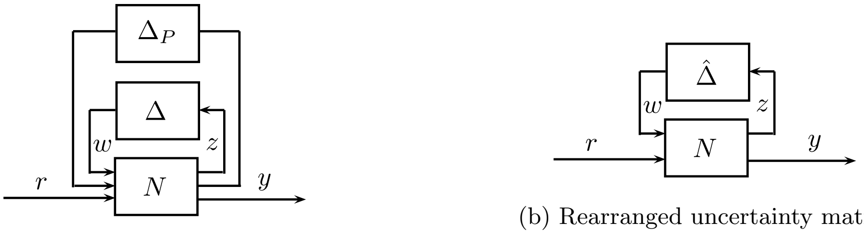
A μ based robust performance test can be set up, however we need to modify the original $M - \Delta$ loop as it can be seen in Figure A.4. A new uncertainty block is wrapped around the original perturbed system. The uncertainties can be treated together as one hypermatrix $\hat{\Delta} = \begin{bmatrix} \Delta & 0 \\ 0 & \Delta_P \end{bmatrix}$.

The following theorem states a μ condition on robust performance.

Theorem A.2.2. *Assume that the uncertain system is given by the block diagram in Figure A.4b. Assume further that the system is nominally stable. Then the condition given in Definition A.2.3 holds if and only if*

$$\mu_{\hat{\Delta}}(N(\mathrm{j}\omega)) < 1, \quad \forall \omega\,, \tag{A.7}$$

where the structural singular value is computed with respect to the structure $\hat{\Delta}$.



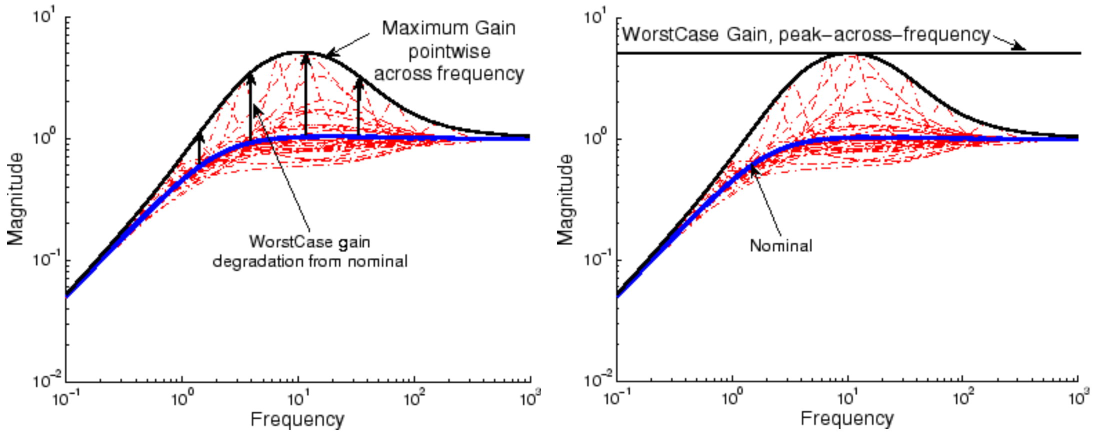
(a) Augmented $N - \Delta$ structure

(b) Rearranged uncertainty matrix

Figure A.4: Block diagram to compute robust stability

See [10] for the proof. Theorem A.2.2 apparaently turned the robust performance question to a robust stability test on the rearranged closed loop.

The *worst case gain* measures the biggest system gain over all the possible uncertainties in $\mathbf{\Delta}$. In this work worst case gain is determined only for SISO systems. Therefore it is measured on the frequency response. Two ways are available to perform the analysis: computing the worst case gain across all the frequencies (Figure A.5a[1]) (this approach is used), or determine only the peak of the gain (Figure A.5b[2]) across all the frequencies and uncertain models.



(a) Max over frequency

(b) Peak over frequency

Figure A.5: Worst case gain calculation methods

[1] `http://www.mathworks.com/help/robust/ref/wcgain.html`

[2] `http://www.mathworks.com/help/robust/ref/wcgain.html`

Appendix B

Analysis of destabilizing p loop

It turned out in Section 3.2.1 that one SISO loop is unstable at the loop-at-a-time margin analyis. We follow the procedure of buliding the lateral/directional baseline controller from a stability point of view in the sequel. Table B.1 tracks how the poles change as the parts of controller are wrapped around the plant. We can follow the process using Figure 2.4.

The linear lateral/directional dynamics of the aircraft is stable, since each pole lies in the negative complex halfplane. When we close the r loop with the proprtional gain K_r, the stability of the loop is preserved. However, by wrapping the p loop around the previous system, one destabilizes the closed loop (therefore a positive pole appears, boldfaced in Table B.1). Closing the ϕ loop with the PI block (K_ϕ) will stabilize the closed loop again. Thus, in the loop-at-a-time analysis, when we open the stabilizing ϕ loop, it results an unstable system.

Loop closed	poles
Open loop	$-0.0138,\ -1.84 \pm 5.28\text{i},\ -16.1$
r loop	$-0.012,\ -2.65 \pm 2.35\text{i},\ -5.73,\ -16.1$
p loop	$\mathbf{0.00193},\ -2.62 \pm 2.11\text{i},\ -5.73,\ 27.1$
ϕ loop	$-0.353,\ -2.62,\ -3.31 \pm 1.81\text{i},\ -5.74,\ 22.8$

Table B.1: Poles of SISO systems: loop broken after plant

The question naturally comes up that do we need this setup in which one loop destabilizes the system, then an other loop stabilizes it as well as ensures the fulfilment of the design goals? If we change the behavior of inner loop lateral/directional dynamics, retuning is necessary in the outer loop controller. In order to avoid this, we try to find controller gains, which reproduce the original inner loop step responses, but without destabilizing the system in any loop.

By choosing $K_p = 0$, $K_{\phi_{prop}} = -0.4414$, $K_{\phi_{int}} = -0.20$, there is only a slight difference between the step responses, as it can be seen in Figure B.1. We keep K_r from the original baseline design.

Nevertheless, if we examine the Bode amplitude diagram of the SISO system with input as elevator disturbance and output as roll rate, it turns out that this choice of gains does not result an appropriate controller. Figure B.2 shows the amplitude plot of the aforementioned systems in case of the open loop system, the original baseline design and the retuned baseline design. As it can be observed, the retuned baseline design has worse disturbance rejection ability than the open loop system around $54\,\frac{\text{rad}}{\text{s}}$.

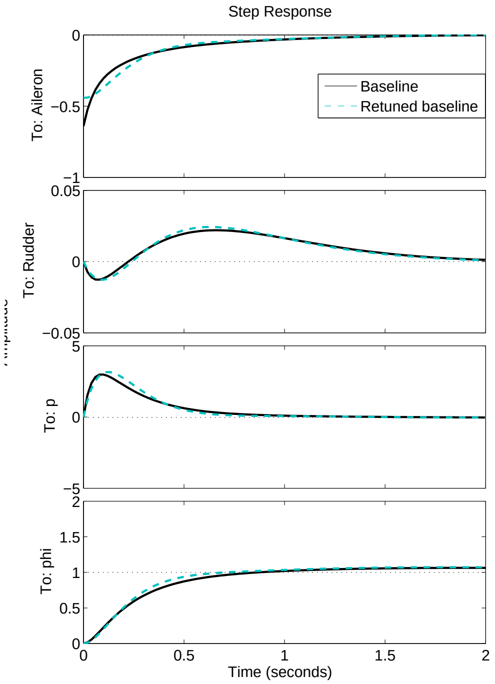


Figure B.1: Step response comparison

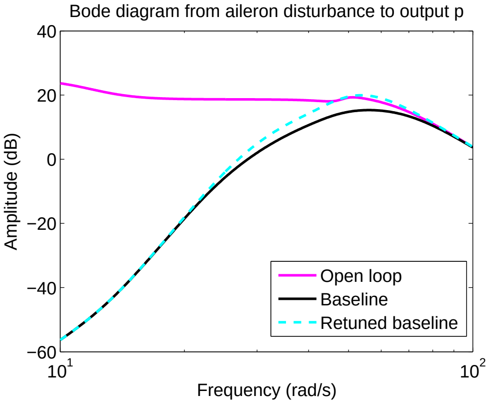


Figure B.2: Bode amplitude diagrams

In the reflection of the results above no change of the baseline controller design is proposed, despite the known destabilzing p loop.

Apart from the analysis above, the unstable pole is at $0.00193\,\frac{\mathrm{rad}}{\mathrm{s}}$. This is extremely slow, so it is not necessary to stabilize it.

Appendix C

RS, RP and worst case perturbations in linear models

This appendix contains tables about the numerical results of robust stability, robust performance and worst case gain analysis of linear systems.

Table C.1: SISO plants in robustness analysis of lateral/directional dynamics - ϕ reference to output

Input	Output	Uncertainty	Controller	RS lb[a]	RS ub[b]	RS pf[c]	RP lb[a]	RP ub[b]	RP pf[c]	WCG lb[a]	WCG ub[b]	WCG pf[c]
ϕ_{ref}	p	Diagonal	Baseline	3.1578	3.1578	0.6220	0.2544	0.2544	10.0694	4.5106	4.5119	10.6421
ϕ_{ref}	p	Diagonal	LQR	2.5646	2.5646	2.1396	0.4496	0.4496	2.8213	2.6659	2.6661	2.8738
ϕ_{ref}	ϕ	Diagonal	Baseline	Same as with output p			0.8913	0.8917	0.7832	1.1312	1.1314	0.8127
ϕ_{ref}	ϕ	Diagonal	LQR	Same as with output p			0.9077	0.9077	1.7149	1.1266	1.1269	1.7958
ϕ_{ref}	p	Full block	Baseline	2.2127	2.2127	0.6821	0.2542	0.2542	10.0694	4.5220	4.5243	10.4477
ϕ_{ref}	p	Full block	LQR	2.1846	2.1846	2.1005	0.4432	0.4432	2.8213	2.8134	2.8136	2.8738
ϕ_{ref}	ϕ	Full block	Baseline	Same as with output p			0.8713	0.8713	0.8749	1.1706	1.1709	0.9332
ϕ_{ref}	ϕ	Full block	LQR	Same as with output p			0.8846	0.8846	1.8124	1.1767	1.1770	1.9155

[a] lower bound

[b] upper bound

[c] peak frequency $\left(\frac{\text{rad}}{\text{s}}\right)$

Table C.2: SISO plants in robustness analysis of lateral/directional dynamics - aileron disturbace to output

Input	Output	Uncertainty	Controller	RS lb[a]	RS ub[b]	RS pf[c]	RP lb[a]	RP ub[b]	RP pf[c]	WCG lb[a]	WCG ub[b]	WCG pf[c]
Ail. dist.	p	Diagonal	Baseline	3.1578	3.1578	0.6220	0.1658	0.1658	9.9770	7.0448	7.0450	10.6421
Ail. dist.	p	Diagonal	LQR	2.5646	2.5646	2.1524	0.1013	0.1013	5.4127	12.8258	12.8263	5.1688
Ail. dist.	ϕ	Diagonal	Baseline	Same as with output p			0.6152	0.6153	1.0618	1.6853	1.6855	1.1325
Ail. dist.	ϕ	Diagonal	LQR	Same as with output p			0.3586	0.3586	2.8384	3.4144	3.4151	2.9382
Ail. dist.	p	Full block	Baseline	2.2127	2.2127	0.6821	0.1657	0.1657	9.9770	7.0625	7.0643	10.5444
Ail. dist.	p	Full block	LQR	2.1846	2.1846	2.1033	0.1009	0.1009	5.4127	13.2045	16.2073	4.9931
Ail. dist.	ϕ	Full block	Baseline	Same as with output p			0.6053	0.6053	1.1016	1.7574	1.7577	1.2080
Ail. dist.	ϕ	Full block	LQR	Same as with output p			0.3357	0.3357	2.9045	3.5531	3.5535	2.9723

[a] lower bound

[b] upper bound

[c] peak frequency $\left(\frac{\text{rad}}{\text{s}}\right)$

Controller	Uncertainty	Output	T_1	T_2	T_3	T_4	A_1	A_2	A_3	A_4
Baseline	Elevator	q	0.01	N/A	N/A	N/A	0.3000	N/A	N/A	N/A
		θ	0.01	N/A	N/A	N/A	0.3000	N/A	N/A	N/A
	Diagonal ail. & rud.	p	0.1000	0.0100	N/A	N/A	0.3000	0.3000	N/A	N/A
		ϕ	0.2570	8.1416	N/A	N/A	0.3000	-0.3000	N/A	N/A
	Full block ail. & rud.	p	0.0100	0.3310	0.0100	0.1395	0.3000	0.3000	0.3000	0.3000
		ϕ	0.0396	0.5855	0.0100	0.4504	0.3000	-0.3000	0.3000	0.1335
LQR	Diagonal ail. & rud.	p	0.0100	0.0100	N/A	N/A	0.3000	0.3000	N/A	N/A
		ϕ	0.0100	0.0100	N/A	N/A	-0.3000	-0.3000	N/A	N/A
	Full block ail. & rud.	p	0.0100	8.7898	1.1349	6.9911	-0.3000	-0.2895	0.3000	0.2094
		ϕ	0.0100	0.0100	0.0100	0.0100	-0.3000	-0.3000	-0.3000	-0.3000

Table C.3: Worst case perturbations in linear systems - angle pattern inputs

Appendix D

Worst case perturbations in nonlinear model

This appendix contains tables about the numerical results of worst case gain analysis of nonlinear systems.

D.1 Angle pattern inputs

Controller	Uncertainty	Output	T_1	T_2	T_3	T_4	A_1	A_2	A_3	A_4
Baseline	Elevator	q	0.0121	N/A	N/A	N/A	0.3000	N/A	N/A	N/A
		θ	0.0289	N/A	N/A	N/A	0.3000	N/A	N/A	N/A
	Diagonal ail. & rud.	p	0.1000	0.0429	N/A	N/A	0.3000	0.3000	N/A	N/A
		ϕ	0.0100	0.0882	N/A	N/A	0.3000	0.3000	N/A	N/A
	Full block ail. & rud.	p	0.1685	0.0494	0.3612	0.0100	0.3000	0.3000	0.1232	0.3000
		ϕ	7.8083	1.7334	1.6544	4.4087	-0.2335	0.3000	-0.3000	0.3000
LQR	Diagonal ail. & rud.	p	0.0100	0.0403	N/A	N/A	0.3000	0.3000	N/A	N/A
		ϕ	0.0100	0.0100	N/A	N/A	-0.3000	-0.3000	N/A	N/A
	Full block ail. & rud.	p	0.0100	0.1736	0.2199	0.0100	0.3000	0.3000	0.3000	0.3000
		ϕ	0.0100	0.0100	0.0100	0.0100	-0.3000	-0.3000	-0.3000	-0.3000

Table D.1: Worst case perturbations in nonlinear systems- angle pattern inputs

D.2 Sinusoidal inputs

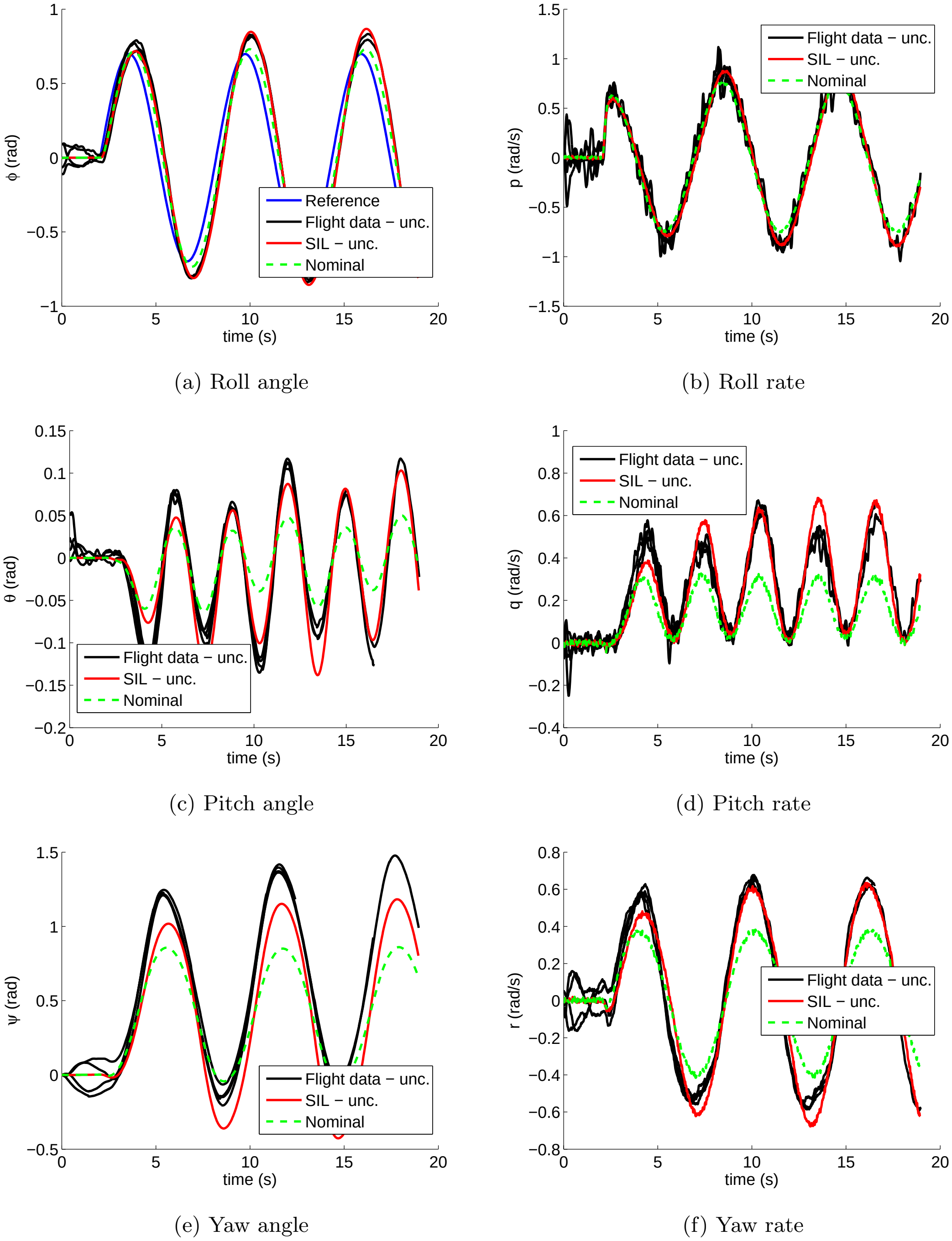
Controller	Uncertainty	Output	T_1	T_2	T_3	T_4	A_1	A_2	A_3	A_4
Baseline	Elevator	q	5.8763	N/A	N/A	N/A	-0.3000	N/A	N/A	N/A
		θ	0.01	N/A	N/A	N/A	0.3000	N/A	N/A	N/A
	Diagonal ail. & rud.	p	6.3247	0.0951	N/A	N/A	-0.2631	0.3000	N/A	N/A
		ϕ	0.0100	0.0294	N/A	N/A	0.3000	-0.3000	N/A	N/A
	Full block ail. & rud.	p								
		ϕ	*	*	*	*	*	*	*	*
LQR	Diagonal ail. & rud.	p	0.0100	0.0294	N/A	N/A	0.3000	-0.3000	N/A	N/A
		ϕ	0.0100	0.0294	N/A	N/A	0.3000	-0.3000	N/A	N/A
	Full block ail. & rud.	p	0.0898	0.5090	0.1599	0.0100	0.2994	0.2292	0.2999	0.3000
		ϕ	5.3451	5.7149	1.9375	0.1854	0.2953	0.2894	0.2957	0.2985

Table D.2: Worst case perturbations in nonlinear systems - sinusoidal inputs

Appendix E

Flight test results

Flight data results are collected in this appendix. Each subsection contains plots for one experiment.



(a) Roll angle

(b) Roll rate

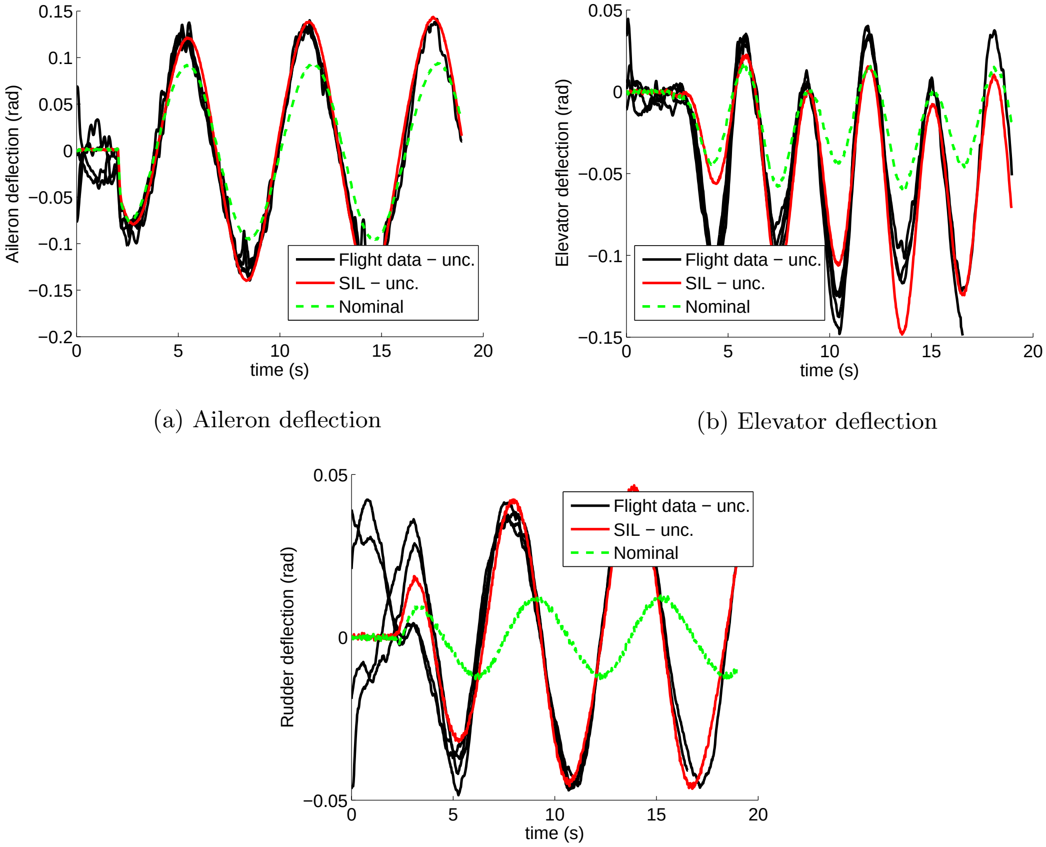
(c) Pitch angle

(d) Pitch rate

(e) Yaw angle

(f) Yaw rate

Figure E.1: Responses in experiment 1

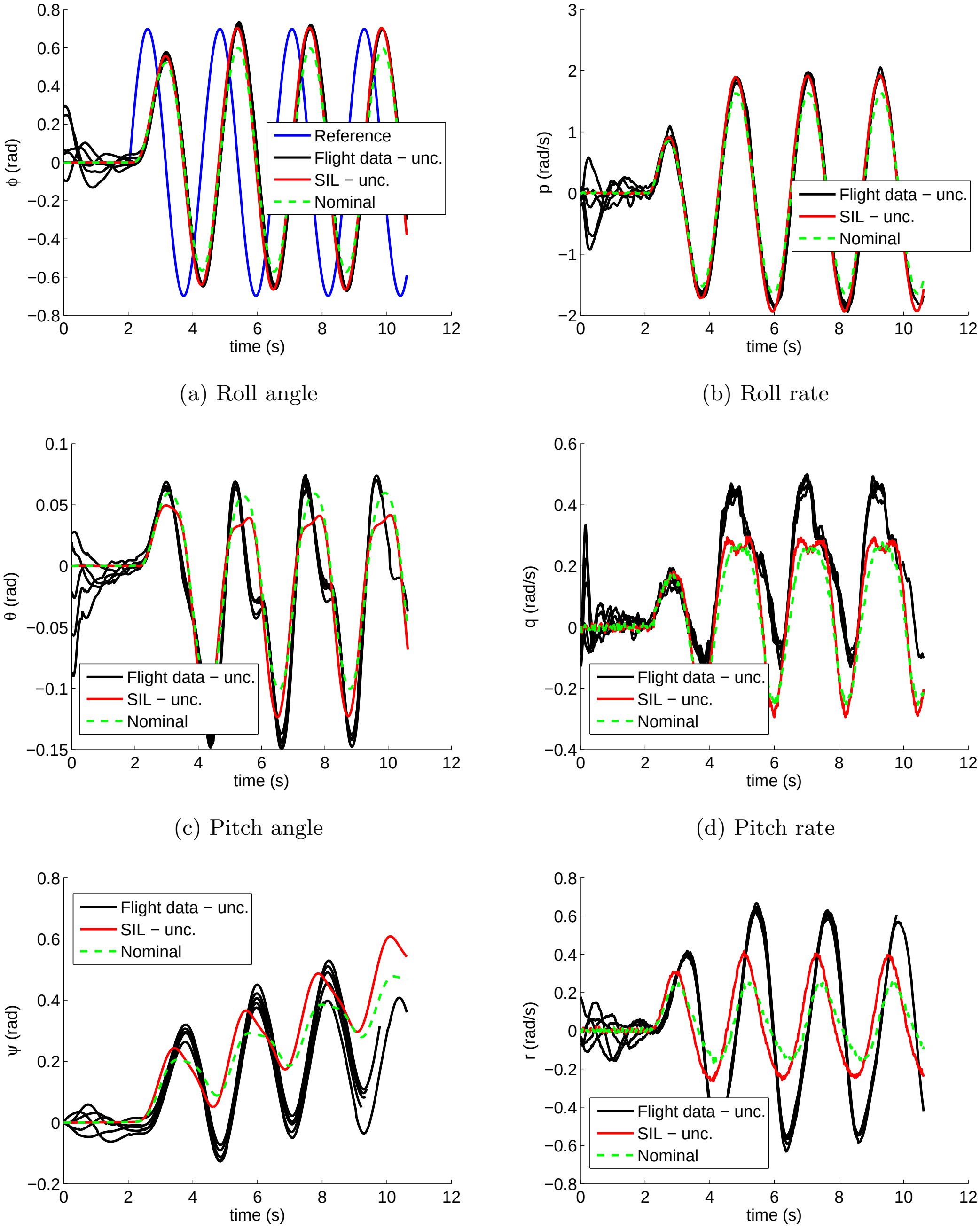


(a) Aileron deflection

(b) Elevator deflection

(c) Rudder deflection

Figure E.2: Surface commands in experiment 1



(a) Roll angle

(b) Roll rate

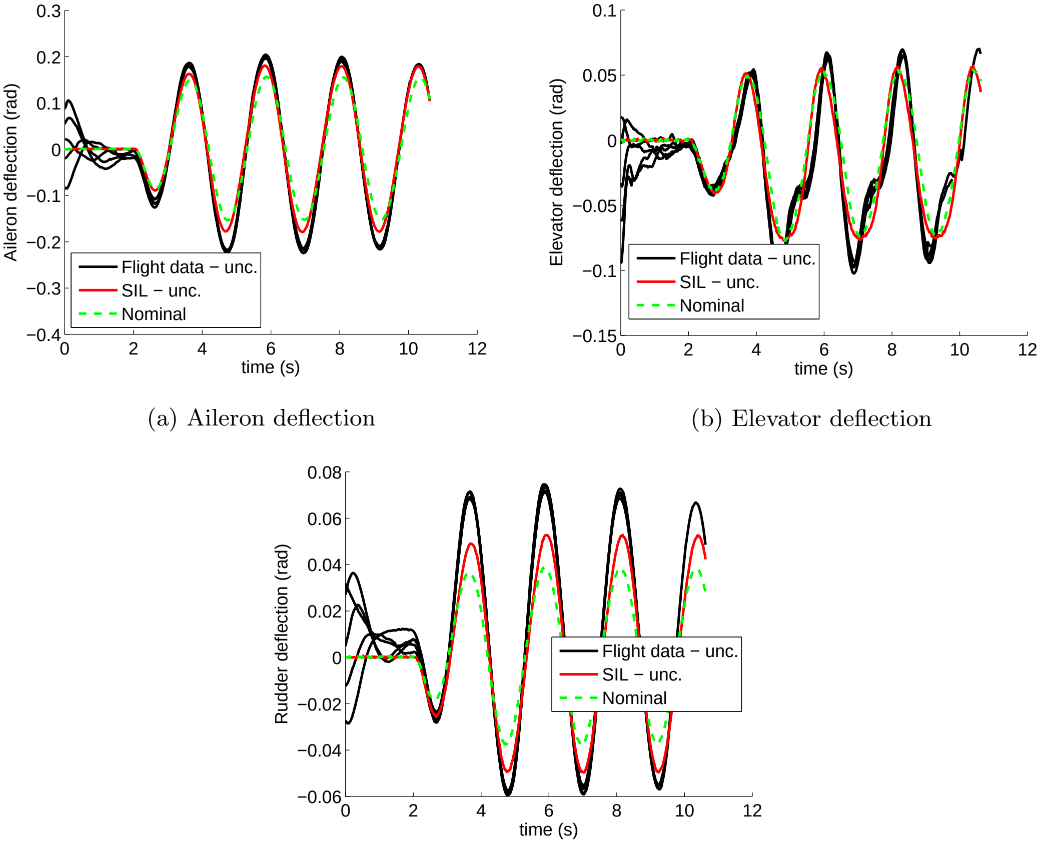
(c) Pitch angle

(d) Pitch rate

(e) Yaw angle

(f) Yaw rate

Figure E.3: Responses in experiment 2

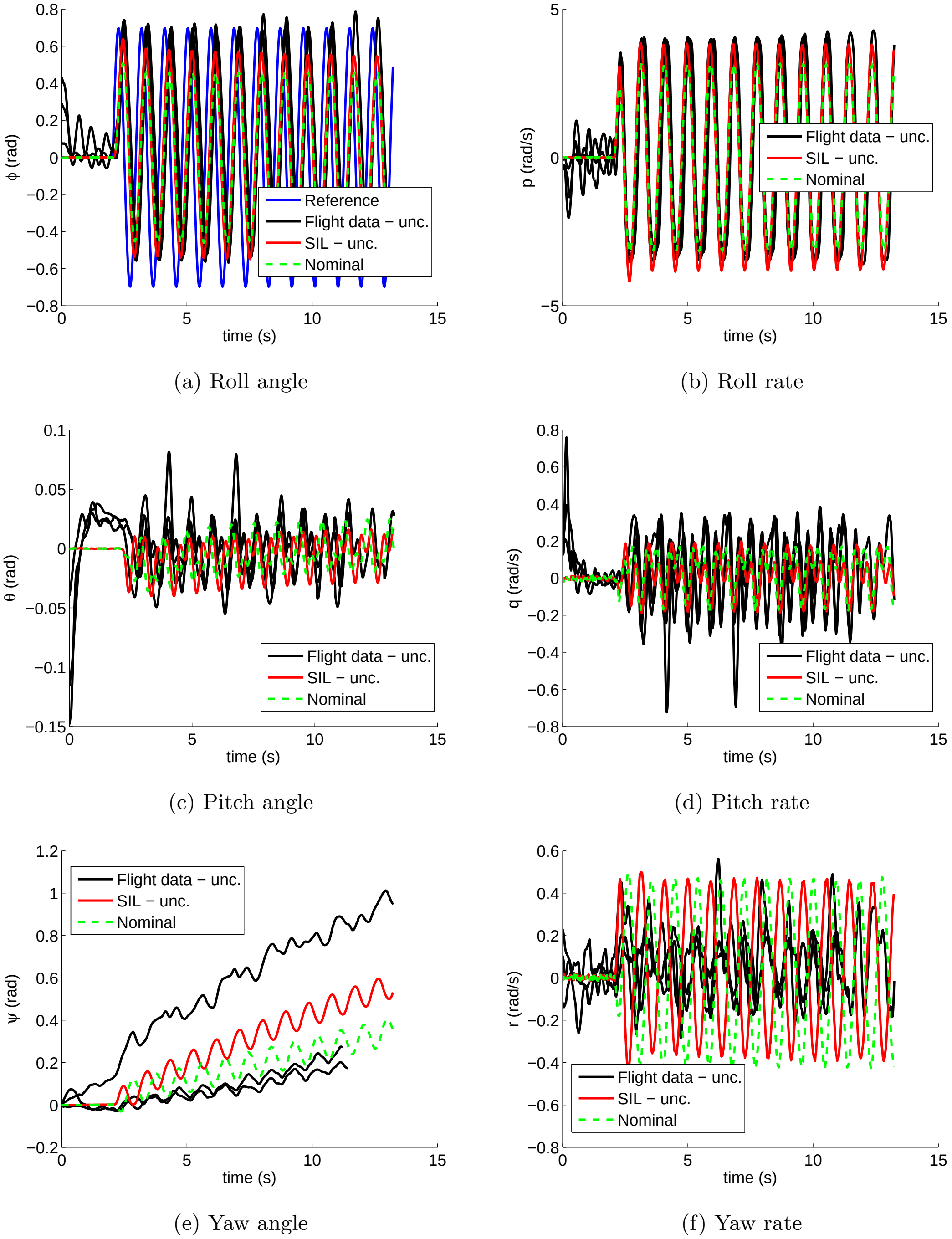


(a) Aileron deflection

(b) Elevator deflection

(c) Rudder deflection

Figure E.4: Surface commands in experiment 2



(a) Roll angle

(b) Roll rate

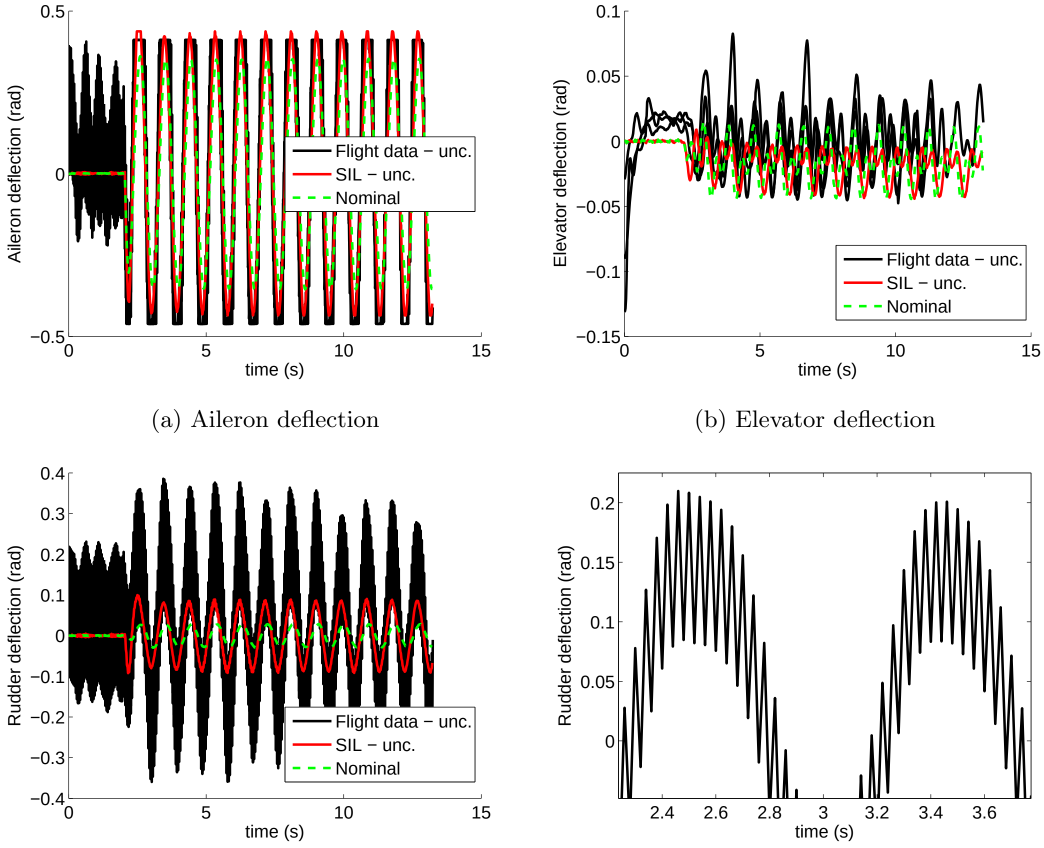
(c) Pitch angle

(d) Pitch rate

(e) Yaw angle

(f) Yaw rate

Figure E.5: Responses in experiment 3



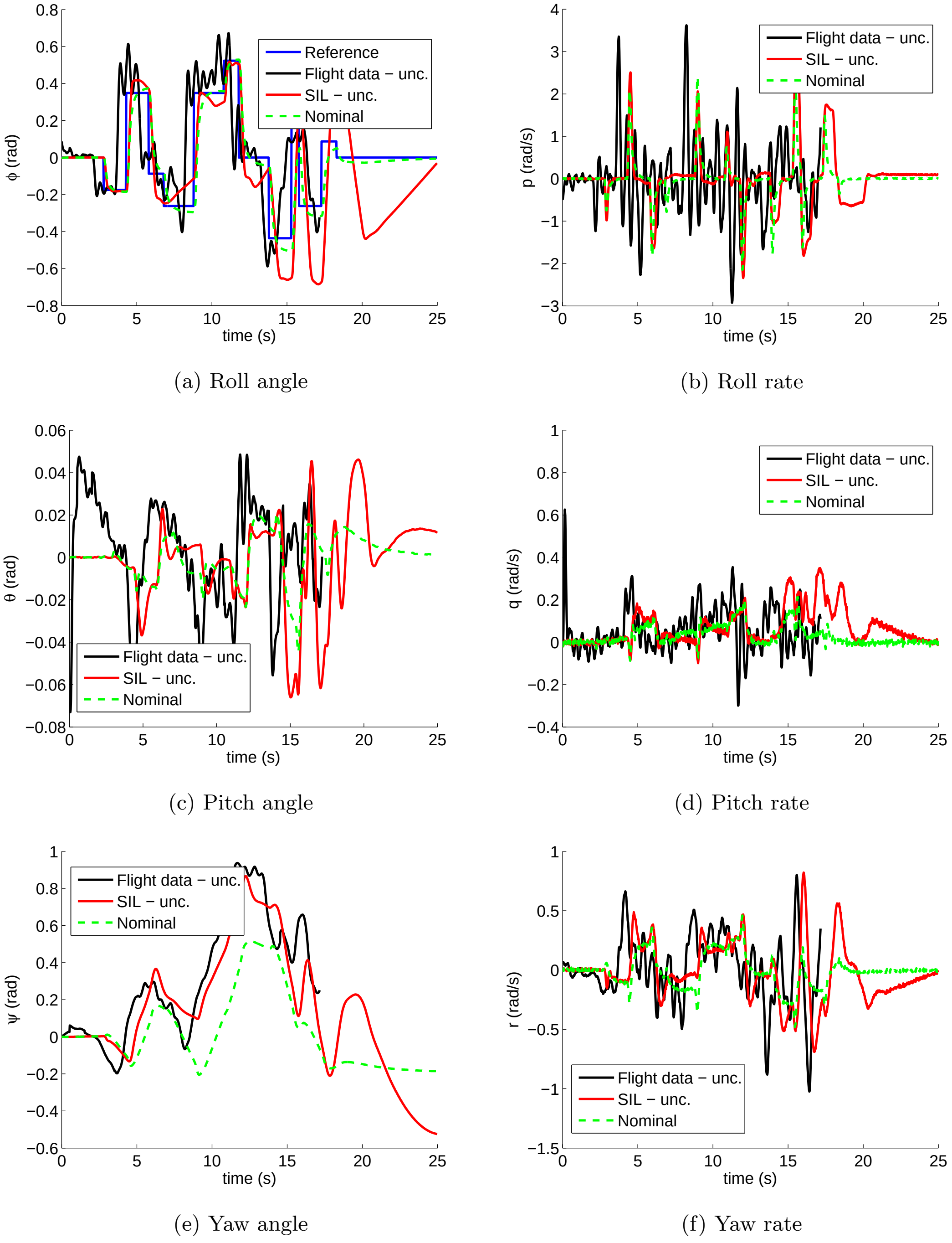
(a) Aileron deflection

(b) Elevator deflection

(c) Rudder deflection

(d) Rudder deflection - closer look

Figure E.6: Surface commands in experiment 3



(a) Roll angle

(b) Roll rate

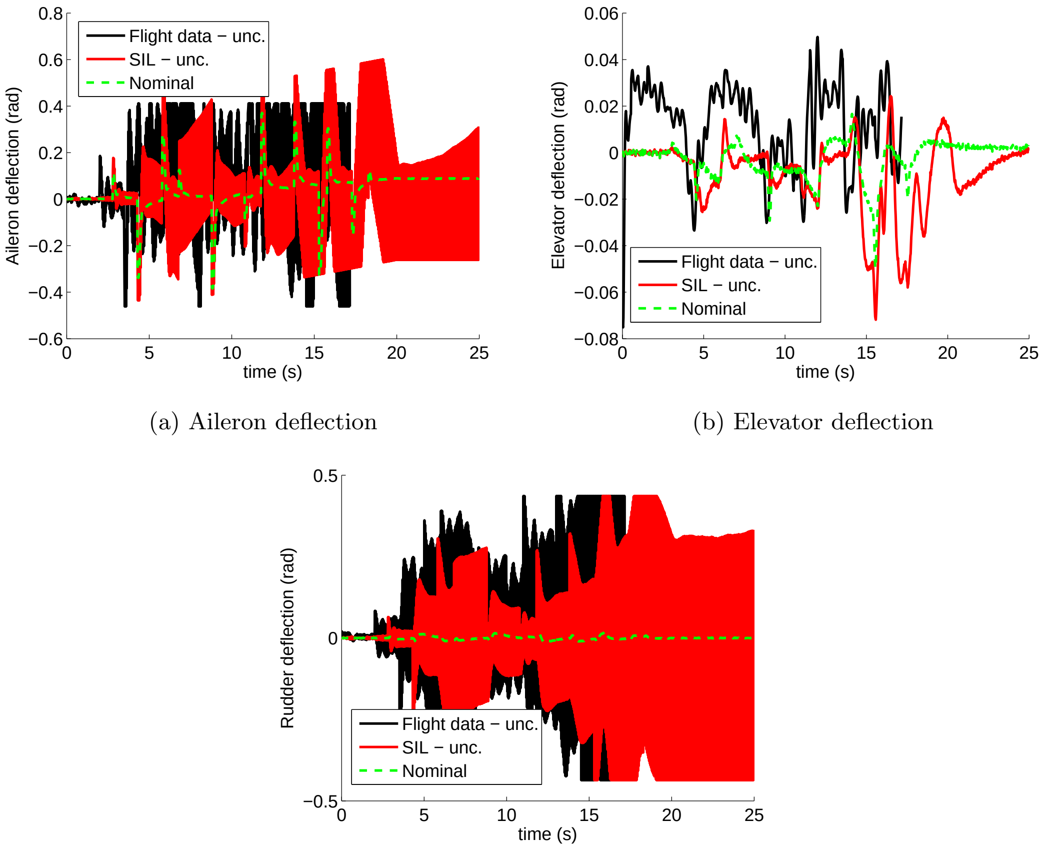
(c) Pitch angle

(d) Pitch rate

(e) Yaw angle

(f) Yaw rate

Figure E.7: Responses in experiment 4

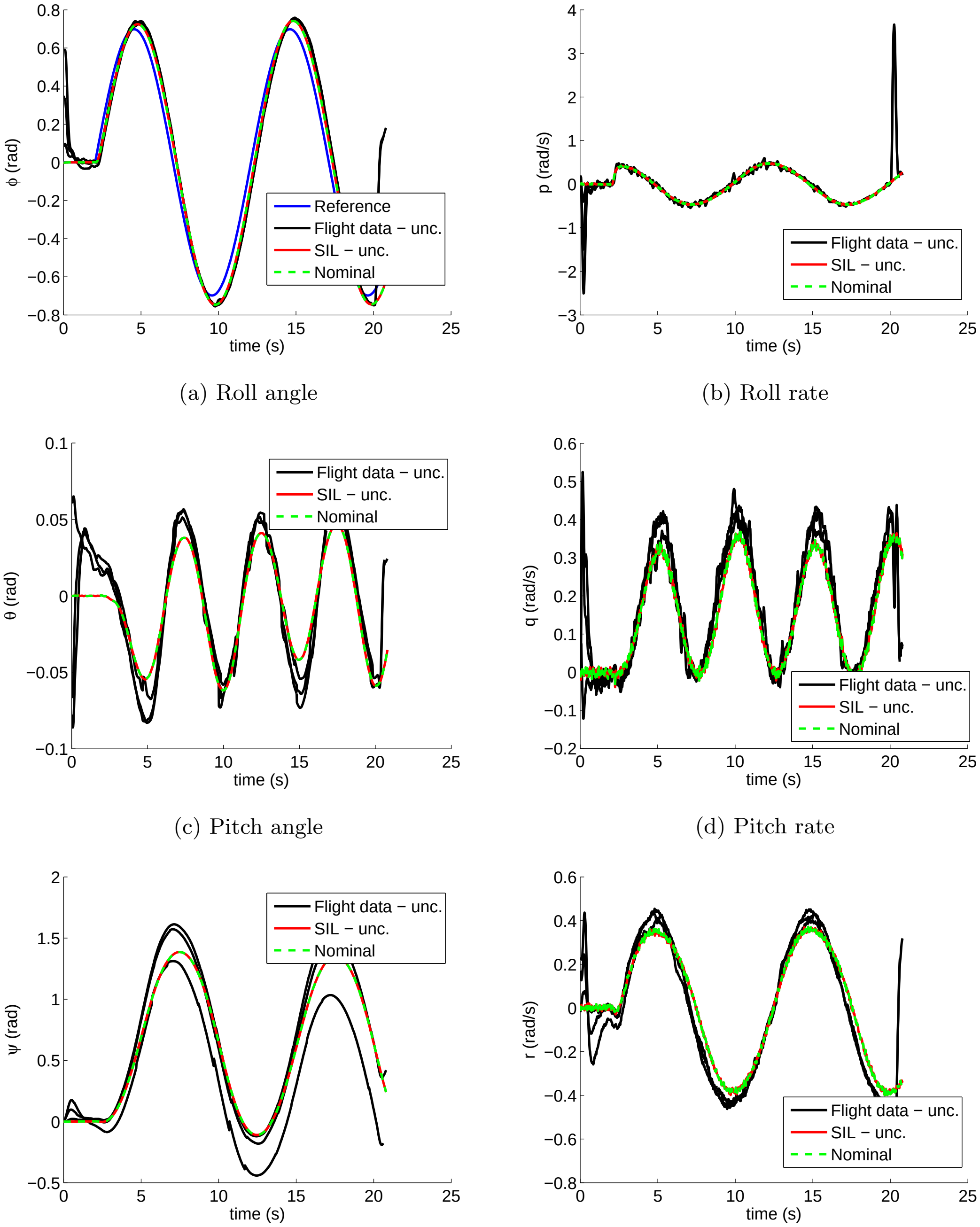


(a) Aileron deflection

(b) Elevator deflection

(c) Rudder deflection

Figure E.8: Surface commands in experiment 4



(a) Roll angle

(b) Roll rate

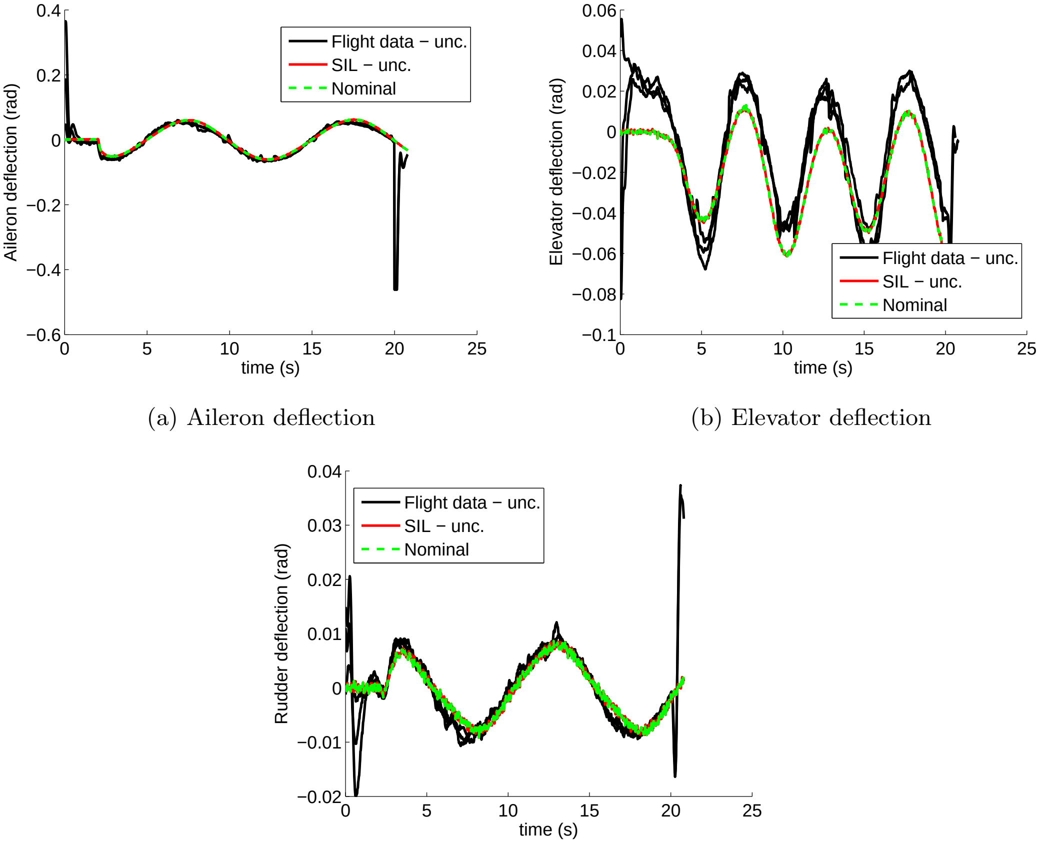
(c) Pitch angle

(d) Pitch rate

(e) Yaw angle

(f) Yaw rate

Figure E.9: Responses in experiment 5

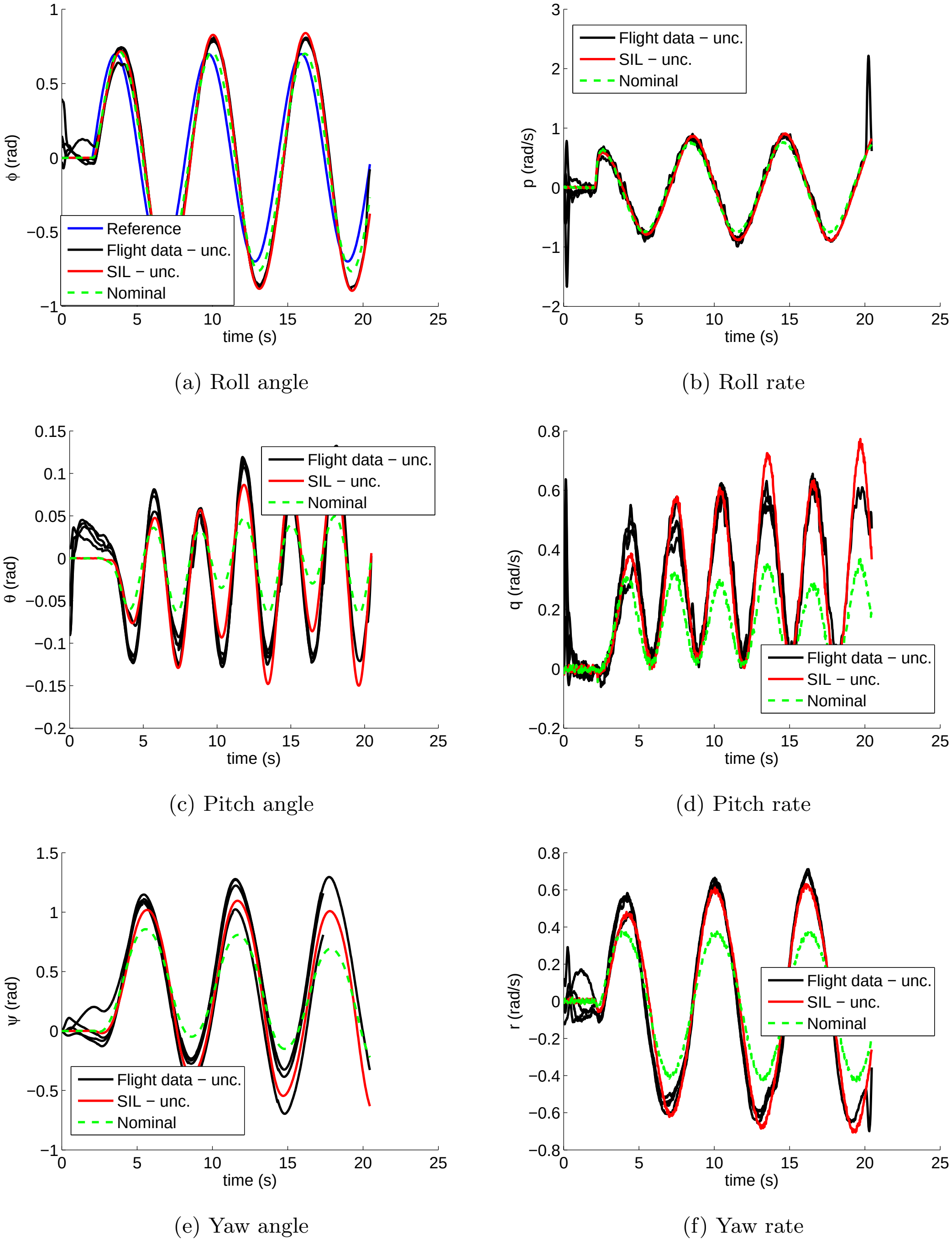


(a) Aileron deflection

(b) Elevator deflection

(c) Rudder deflection

Figure E.10: Surface commands in experiment 5



(a) Roll angle

(b) Roll rate

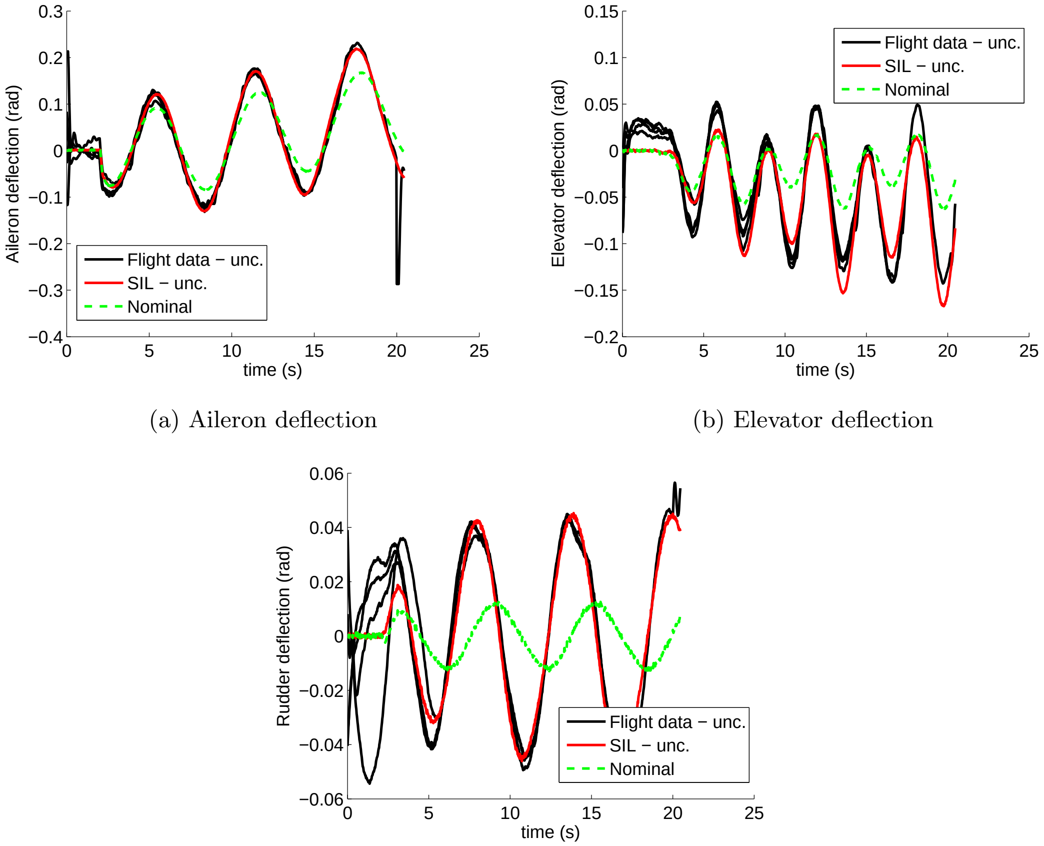
(c) Pitch angle

(d) Pitch rate

(e) Yaw angle

(f) Yaw rate

Figure E.11: Responses in experiment 6



(a) Aileron deflection

(b) Elevator deflection

(c) Rudder deflection

Figure E.12: Surface commands in experiment 6