Direct Lift Control of Fighter Aircraft

Markus Åstrand & Philip Öhrn
Master of Science Thesis in Electrical Engineering

Direct Lift Control of Fighter Aircraft
Markus Åstrand & Philip Öhrn
LiTH-ISY-EX--19/5214--SE

Supervisors:  
Angela Fontan, M.Sc.  
ISY, Linköping university
Peter Rosander, Lic.  
Saab Aeronautics
Emil Johansson, M.Sc.  
Saab Aeronautics

Examiner:  
Johan Löfberg, Assoc. Prof.  
ISY, Linköping university

Division of Automatic Control  
Department of Electrical Engineering  
Linköping University  
SE-581 83 Linköping, Sweden

Copyright © 2019 Markus Åstrand & Philip Öhrn
"Skönt att ha flygplanet i sin hand igen..."
Abstract

Direct lift control for aircraft has been around in the aeronautical industry for decades but is mainly used in commercial aircraft with dedicated direct lift control surfaces. The focus of this thesis is to investigate if direct lift control is feasible for a fighter aircraft, similar to Saab JAS 39 Gripen, without dedicated control surfaces.

The modelled system is an aircraft that is inherently unstable and contains nonlinearities both in its aerodynamics and in the form of limited control surface deflection and deflection rates. The dynamics of the aircraft are linearised around a flight case representative of a landing scenario. Direct lift control is then applied to give a more immediate relation from pilot stick input to change in flight path angle while also preserving the pitch attitude.

Two different control strategies, linear quadratic control and model predictive control, were chosen for the implementation. Since fighter aircraft are systems with fast dynamics it was important to limit the computational time. This constraint motivated the use of specialised methods to speed up the optimisation of the model predictive controller.

Results from simulations in a nonlinear simulation environment supplied by Saab, as well as tests in high-fidelity flight simulation rigs with a pilot, proved that direct lift control is feasible for the investigated fighter aircraft. Sufficient control authority and performance when controlling the flight path angle were observed. Both developed controllers have their own advantages and which strategy is the most suitable depends on what the user prioritises. Pilot workload during landing as well as precision at touch down were deemed similar to conventional control.
Acknowledgments

We would like to thank our supervisors at Saab, Peter Rosander and Emil Johanson for their help and technical expertise in control theory and flight mechanics. Their ideas and advice have been invaluable to this thesis.

Our supervisor at Linköping university, Angela Fontan, deserves a special thanks for proofreading this report and for her quick response when providing feedback.

Lastly, we would like to express our gratitude to Erik Bodin at Saab, for his help with the simulators and making the landing study possible, and to test pilot Mikael Seidl for his evaluation of the controllers.

Linköping, May 2019
Markus Åstrand & Philip Öhrn
## Contents

<table>
<thead>
<tr>
<th>List of Figures</th>
<th>xi</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of Tables</td>
<td>xiv</td>
</tr>
<tr>
<td>Notation</td>
<td>xv</td>
</tr>
</tbody>
</table>

### 1 Introduction
1.1 Problem formulation .................. 1
1.2 Methodology ............................ 2
1.3 Related work ........................... 2
1.4 Outline ................................. 2

### 2 Aircraft theory
2.1 Flight mechanics ........................ 5
2.2 Direct lift control ..................... 7
2.3 System description ..................... 7

### 3 Control strategies
3.1 Linear quadratic control ............. 9
  3.1.1 Stability ........................... 10
3.2 Model predictive control ............... 11
  3.2.1 Problem formulation ................. 11
  3.2.2 Compact description ................. 12
  3.2.3 Reference tracking ................. 13
  3.2.4 Integral action ..................... 13
3.3 Fast MPC ................................ 14
  3.3.1 Primal barrier interior-point method 15
  3.3.2 Infeasible start Newton method .......... 16
  3.3.3 Further approximations ............. 17

### 4 Implementation
4.1 LQ controller ........................... 19
  4.1.1 Baseline controller ................. 19
4.1.2 Feedforward of reference ........................................ 20
4.1.3 Anti-windup compensator ....................................... 21
4.2 Fast MPC ......................................................................... 22
4.2.1 Matrix definitions ..................................................... 22
4.2.2 Constraints ............................................................... 23
4.2.3 Integral action .......................................................... 24
4.2.4 Tuning parameters ..................................................... 25
4.3 Pilot interface ............................................................... 25
4.3.1 Delta mode ............................................................... 25
4.3.2 Set-point mode ........................................................ 26
4.4 Simulation setup ........................................................... 27
4.4.1 Modifications for DLC ............................................... 27
5 Results ............................................................................. 29
5.1 Direct lift control ........................................................... 29
5.1.1 Delta mode ............................................................... 29
5.1.2 Set-point mode ........................................................ 31
5.1.3 Disturbance rejection ................................................ 33
5.1.4 Stability ................................................................. 35
5.2 Comparison against conventional control ....................... 37
5.2.1 Transient response .................................................... 37
5.2.2 Landing study .......................................................... 38
5.3 Additional plots ............................................................. 42
5.4 Landing study - additional plots ..................................... 46
5.5 Results from LQ extensions ........................................... 49
5.6 MPC prediction horizons ................................................. 51
6 Discussion and conclusions ............................................... 53
6.1 Direct lift control ........................................................... 53
6.1.1 Feasibility .............................................................. 53
6.1.2 Comparison against conventional control .................... 54
6.2 Comparison of implemented controllers ......................... 55
6.2.1 Performance ........................................................... 55
6.2.2 Robustness and stability .......................................... 55
6.2.3 Complexity ............................................................. 56
6.3 Practical observations .................................................... 57
6.4 Future work ................................................................. 58
Bibliography ......................................................................... 59
# List of Figures

2.1 Aircraft body fixed frame with linear velocities, angular rates and control surfaces. Image provided by Saab. ............................... 5

2.2 Illustration of the aircraft orientation angles $\phi$, $\theta$ and $\psi$, the aero-dynamic angles $\alpha$ and $\beta$, and the angular rates $p$, $q$ and $r$. In the figure, all angles are positive. Figure and caption from [10, Fig 2.2]. 6

2.3 Block diagram of the controlled system. ................................. 8

3.1 The disk gain margin $d g_m$ and disk phase margin $d \phi_m$ can be seen together with the classical gain and phase margins $g_m$ and $\phi_m$. . . . 10

3.2 Basic structure of MPC [4, Fig. 1.2]. ................................. 11

4.1 Block diagram of the state feedback. ................................. 20

4.2 Full system block diagram. Note that the control signal saturations are normally included in $G_{A/C}$ but have been extracted for illustrative purposes. ................................. 22

4.3 Stick input in the two different modes. The grey circles indicate different stick deflections. Note that hard stop is not used in set-point mode. ................................. 26

4.4 Overview of the simulation system. ................................. 27

4.5 Moving average smoothing filter for $\alpha$. Soft stop aft stick deflection is applied at $t = 10$. ................................. 28

5.1 Step response from a maximum aft stick deflection at time $t = 10$ in delta mode. ................................. 30

5.2 Response from a maximum aft stick deflection at time $t = 10$ in set-point mode. ................................. 31

5.3 Steady state behaviour in very light turbulence. ................................. 33

5.4 Response from a horizontal wind disturbance. ................................. 34

5.5 Response from a vertical wind disturbance. ................................. 35

5.6 Disk margin of the LQ controller. Allowed gain variation up to $\pm 5.35$ dB and phase variation up to $33.26^\circ$. Stability is guaranteed for all variations in the input inside the ellipse. ................................. 36
5.7 Nyquist diagrams of the system at each input channel. Classical stability margins can be extracted for each channel. 36
5.8 Transient response of maximum aft stick deflection of the two DLC controllers and conventional control. 37
5.9 Control surface response of maximum aft stick deflection of, from left to right: MPC, LQ and conventional control. 37
5.10 Altitude and flight path angle $\gamma$ during the first LQ test flight. 38
5.11 Altitude and flight path angle $\gamma$ during the first MPC test flight. 39
5.12 Altitude and flight path angle $\gamma$ during the first conventional test flight. 39
5.13 Stick deflection during the LQ test flights. From left to right: test flight 1, 2 and 3. 40
5.14 Stick deflection during the MPC test flights. From left to right: test flight 1, 2 and 3. 40
5.15 Stick deflection during the conventional test flights. From left to right: test flight 1, 2 and 3. 40
5.16 Touch down points for all test flights. The black X marks the desired touch down spot. 41
5.17 Step response from a maximum forward stick deflection at time $t = 10$ in delta mode. 42
5.17 Step response from a maximum forward stick deflection at time $t = 10$ in delta mode. 43
5.18 Response from a maximum forward stick deflection at time $t = 10$ in set-point mode. 43
5.19 Steady state behaviour in light turbulence. 44
5.20 Step response from a maximum aft stick deflection at time $t = 10$ during slow flight in delta mode. 45
5.21 Altitude and flight path angle $\gamma$ during the second LQ test flight. 46
5.22 Altitude and flight path angle $\gamma$ during the second MPC test flight. 47
5.23 Altitude and flight path angle $\gamma$ during the second conventional test flight. 47
5.24 Altitude and flight path angle $\gamma$ during the third LQ test flight. 48
5.25 Altitude and flight path angle $\gamma$ during the third MPC test flight. 48
5.26 Altitude and flight path angle $\gamma$ during the third conventional test flight. 49
5.27 Transient response with and without the feedforward extension of the LQ controller. 49
5.28 Commanded control signals before and after amplitude and rate saturations without anti-windup compensation in the LQ controller. A step that saturates the canards is applied at time $t = 10$ and removed at $t = 20$. Notice the delay due to integrator windup. 50
5.29 Commanded control signals before and after amplitude and rate saturations with anti-windup compensation in the LQ controller. Once again a step that saturates the canards is applied at time $t = 10$ and removed at $t = 20$. Now the unsaturated control signal follows the saturated. 50
5.30 Step response with prediction horizon $N = 2$. ................. 51
5.31 Step response with prediction horizon $N = 8$. ................. 51
## List of Tables

5.1 Time constant $\tau$ and rise time for $\gamma$ with different stick inputs in *delta mode*. Positive difference corresponds to the MPC being faster than the LQ. ................................................................. 31

5.2 Overshoot and settling time for $\gamma$ with different stick inputs in *set-point mode*. Positive difference corresponds to the MPC being faster than the LQ. ................................................................. 32

5.3 Time constant and rise time for $\gamma$ with different stick inputs in *set-point mode*. Positive difference corresponds to the MPC being faster than the LQ. ................................................................. 32

5.4 Mean and variance of $\gamma$ for different levels of turbulence. ....... 34

6.1 Design parameters for the LQ controller. ............................... 56

6.2 Design parameters for the MPC controller. ............................... 57
## Notation

### Notations

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gamma$</td>
<td>Flight path angle</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Angle of attack</td>
</tr>
<tr>
<td>$\phi$</td>
<td>Roll angle</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Pitch angle</td>
</tr>
<tr>
<td>$\psi$</td>
<td>Yaw angle</td>
</tr>
<tr>
<td>$p$</td>
<td>Roll angle rate</td>
</tr>
<tr>
<td>$q$</td>
<td>Pitch angle rate</td>
</tr>
<tr>
<td>$r$</td>
<td>Yaw angle rate</td>
</tr>
<tr>
<td>$\delta_c$</td>
<td>Canard deflection angle</td>
</tr>
<tr>
<td>$\delta_e$</td>
<td>Elevator deflection angle</td>
</tr>
<tr>
<td>$G$</td>
<td>Aircraft dynamics</td>
</tr>
<tr>
<td>$G_{act}$</td>
<td>Actuator dynamics</td>
</tr>
<tr>
<td>$G_{A/C}$</td>
<td>Complete system dynamics</td>
</tr>
</tbody>
</table>

### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARES</td>
<td>Aircraft Rigid-Body Engineering Simulation</td>
</tr>
<tr>
<td>DLC</td>
<td>Direct Lift Control</td>
</tr>
<tr>
<td>LQ</td>
<td>Linear Quadratic</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple Input Multiple Output</td>
</tr>
<tr>
<td>MPC</td>
<td>Model Predictive Control</td>
</tr>
<tr>
<td>QP</td>
<td>Quadratic Programming</td>
</tr>
</tbody>
</table>
The possibility to directly control the flight path angle of an aircraft has proven to be beneficial for precision flight cases such as aircraft carrier landings and in-flight refuelling. Normally the flight path angle is controlled through changes in moment, making it difficult to apply small changes in flight path angle while maintaining a desired pitch angle. This problem can be overcome by using Direct Lift Control (DLC) to control the lift force and get a more immediate relation from pilot input to the flight path angle, while keeping changes in the pitch angle as close to zero as possible. For this purpose, many commercial aircraft use dedicated control surfaces designed to give little to no change in moment. For aircraft that are not equipped with these dedicated surfaces the use of multiple sets of conventional control surfaces can be adopted to control the lift while cancelling the pitching moment.

1.1 Problem formulation

The purpose of this master thesis is to investigate how DLC can be implemented in a fighter aircraft that lacks dedicated direct lift control surfaces. Two control strategies are analysed and compared: linear quadratic control (LQ) and model predictive control (MPC). The questions in focus are:

- Does DLC without dedicated control surfaces generate enough lift to be feasible as an alternative to conventional landing methods for fighter aircraft?
- How does the performance of landing with direct lift compare to a conventional landing?
- Which control strategy is best suited for implementing direct lift in a fighter aircraft with regards to performance, robustness and complexity?
1.2 Methodology

To test the feasibility of DLC using conventional control surfaces a simple controller was first implemented with no regards to stability, robustness or overall performance. The resulting controller was used to evaluate whether or not sufficient lift could be generated with a constant pitch attitude.

Using a linearised model from a flight case with conditions similar to those during approach and landing, the two studied controllers (LQ and MPC) were synthesised and tuned offline. The LQ controller was implemented in a step-by-step manner as described in Chapter 4, with gradual improvements. For the MPC controller, most of the functionality was implemented at once, with integral action and approximations to further reduce computational time added as described in Chapter 4.

When the performance of both offline controllers was deemed satisfactory they were implemented in a nonlinear simulation environment, Aircraft Rigid-Body Engineering Simulation (ARES), supplied by Saab. The simulation environment uses the same aerodynamic data and aircraft servo dynamics as ADMIRE [6], a Simulink model of a fighter aircraft similar to JAS 39 Gripen developed by the Swedish Defence Research Agency. The controllers were then tuned again to work better with the nonlinear aircraft model. Test flights in a high fidelity simulator were also conducted to detect non-obvious characteristics that were difficult to observe in sampled data. This also highlighted pilot-in-the-loop aspects.

1.3 Related work

The inspiration for this thesis came from the American "Maritime Augmented Guidance with Integrated Controls for Carrier Approach and Recovery Precision Enabling Technologies" (MAGIC CARPET) control mode. MAGIC CARPET is being implemented in the US Navy’s F/A-18 Super Hornet and EA-18G Growler aircraft to reduce pilot workload during aircraft carrier operations by utilising direct lift and automatic throttle control.

Most of the previous work related to DLC has focused on the effects of dedicated control surfaces. This has provided some justification as to why DLC is a useful control strategy. Conclusions such as the non-minimum phase behaviour of changes in the path angle being eliminated [11] and pilot workload reduction [12] have been shown.

A thesis similar to this work investigated how DLC could be used to reduce passenger discomfort during turbulence: by developing PID feedback controllers using conventional control surfaces and flaps, changes in vertical acceleration during gusts could be reduced [7].

1.4 Outline

This thesis is structured in a manner that gives the reader a brief theoretical background in the relevant fields before presenting the implementation and results.
General theory regarding aircraft and the control strategies investigated in the thesis can be found in Chapters 2 and 3, respectively. Chapter 4 explains how the direct lift controllers were implemented, the pilot interface and how the simulation environment was set up. The results of the thesis are presented in Chapter 5. Finally, a discussion of the results, conclusions and suggestions on future work are given in Chapter 6.
This chapter gives a brief explanation of the theoretical foundation regarding aircraft dynamics used in the thesis. For a more detailed explanation of the different subjects discussed, the reader is referred to the cited sources.

2.1 Flight mechanics

The following information is based on [13, Ch. 2 and 3].

*Figure 2.1:* Aircraft body fixed frame with linear velocities, angular rates and control surfaces. Image provided by Saab.
An aircraft is modelled with several angles and moments. In Figure 2.1 a body-fixed coordinate system \( \{x_b, y_b, z_b\} \) with \( \{u, v, w\} \) as linear velocities about respective body-fixed axis is shown together with the angular velocities \( \{p, q, r\} \) and the velocity vector \( V \). Also shown in the figure are the control surfaces of the aircraft \( \{\delta_c, \delta_{ei}, \delta_{ey}, \delta_r\} \), where the inner elevator \( (\delta_{ei}) \) and the outer elevator \( (\delta_{ey}) \) are moved symmetrically and hence considered to be a single control surface, \( \delta_e \). With these definitions the equations of motion can be derived through forces and moments acting on the aircraft.

The position and orientation of the aircraft from an earth-fixed frame of reference can be described using Euler angles, \( \{\phi, \theta, \psi\} \) - roll, pitch and yaw. These angles are presented in Figure 2.2. Relations between the body-fixed angular velocities, \( \{p, q, r\} \), and earth-fixed Euler rates, \( \{\phi, \theta, \psi\} \), are described in equation (2.1), where \( C_* \) denotes cosine and \( S_* \) sine of the indexed angle.

\[
\begin{bmatrix}
p \\
q \\
r
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & -S_\theta \\
0 & C_\phi & C_\theta S_\phi \\
0 & -S_\phi & C_\theta C_\phi
\end{bmatrix}
\begin{bmatrix}
\dot{\phi} \\
\dot{\theta} \\
\dot{\psi}
\end{bmatrix}
\] (2.1)

Another key component in describing the motion of an aircraft is the angle between the velocity vector \( V \) and a body-fixed reference line. In Figure 2.2 the angle of attack, \( \alpha \), and the flight path angle, \( \gamma \), are shown together with previously described attitude angles, rates and sideslip angle \( \beta \).

Another key component in describing the motion of an aircraft is the angle between the velocity vector \( V \) and a body-fixed reference line. In Figure 2.2 the angle of attack, \( \alpha \), and the flight path angle, \( \gamma \), are shown together with previously described attitude angles, rates and sideslip angle \( \beta \).

**Figure 2.2:** Illustration of the aircraft orientation angles \( \phi, \theta \) and \( \psi \), the aerodynamic angles \( \alpha \) and \( \beta \), and the angular rates \( p, q \) and \( r \). In the figure, all angles are positive. Figure and caption from [10, Fig 2.2].

When the motion is purely longitudinal a simple linear model can be constructed with aerodynamic force and moment derivatives. The short-period approximation model is shown in state space representation in equation (2.2),

\[
\begin{bmatrix}
\Delta \dot{\alpha} \\
\Delta \dot{q}
\end{bmatrix} =
\begin{bmatrix}
\frac{Z_\alpha}{u_0} & 1 \\
M_\alpha + M_\alpha Z_\alpha u_0 & M_q + M_q Z_\alpha
\end{bmatrix}
\begin{bmatrix}
\Delta \alpha \\
\Delta q
\end{bmatrix} +
\begin{bmatrix}
Z_\delta \\
M_\delta + M_\delta Z_\delta
\end{bmatrix}
\begin{bmatrix}
\Delta \delta \\
\Delta \delta_T
\end{bmatrix}
\] (2.2)
where $\delta$ is the control surface deflection and $\delta_T$ is commanded engine thrust. The parameters $Z_*$ and $M_*$ are aerodynamic force and moment derivatives. Values for these derivatives depend on the current flight case and are typically extracted from look-up tables. Finally, $u_0$ is the velocity in the body-fixed $x$-direction around which the linear model is derived. The state space representation in equation (2.2) is later referred to in its compact form

\[
\begin{bmatrix}
\Delta \dot{\alpha} \\
\Delta \dot{q} \\
\Delta \dot{\theta}
\end{bmatrix}
= A_{\alpha q}\begin{bmatrix}
\Delta \alpha \\
\Delta q
\end{bmatrix} + B_{\alpha q}\begin{bmatrix}
\Delta \delta
\end{bmatrix}.
\] (2.3)

In this model the thrust, $\Delta \delta_T$, is omitted and thus the corresponding column in model (2.2) is removed.

### 2.2 Direct lift control

A practical implementation of DLC would be to control the angle of attack, $\Delta \alpha$, to a desired value while keeping the pitch angle, $\Delta \theta$, close to zero; consequently, the flight path angle $\Delta \gamma$ becomes approximately $-\Delta \alpha$. To control $\Delta \theta$ model (2.3) must be extended with a $\Delta \theta$ state. If the aircraft is assumed to be in a configuration as described in Section 2.1 the model can be written as

\[
\begin{bmatrix}
\Delta \dot{\alpha} \\
\Delta \dot{q} \\
\Delta \dot{\theta}
\end{bmatrix}
= \begin{bmatrix}
A_{\alpha q} & 0 \\
0 & 1 \\
0 & 0
\end{bmatrix}\begin{bmatrix}
\Delta \alpha \\
\Delta q \\
\Delta \theta
\end{bmatrix} + \begin{bmatrix}
B_{\alpha q} & 0 \\
0 & 0
\end{bmatrix}\begin{bmatrix}
\Delta \delta_c \\
\Delta \delta_e
\end{bmatrix}.
\] (2.4)

Here $\Delta \delta$ is split into $\Delta \delta_c$ and $\Delta \delta_e$, which represent the deflection angles of the canard and elevator control surfaces, respectively. Model (2.4) is henceforth referred to in its compact form

\[
\dot{x} = Ax + Bu \tag{2.5}
\]

where $x = [\Delta \alpha \quad \Delta q \quad \Delta \theta]^T$ and $u = [\Delta \delta_c \quad \Delta \delta_e]^T$. Note that the $\Delta$ is dropped to simplify the notation later on in the report.

When taking control signal constraints such as saturations into consideration, the desired reference might not be possible to reach if $\Delta \theta = 0$. This is natural since the maximum possible achievable lift force is limited by control surface area and control surface deflection.

### 2.3 System description

The system investigated in this thesis can be represented with aircraft and actuator dynamics together with a rate limiter and a saturation, as depicted in Figure 2.3.
The modelled aircraft $G$ is similar to Saab JAS 39 Gripen and is inherently unstable. The actuator dynamics $G_{\text{act}}$ is modelled as a first order system and the rate limit and saturation are chosen to have values similar to those used in Gripen. The input $u$ and the output $y$ are the commanded deflections for the elevator and canard control surfaces $\delta_c$ and $\delta_e$, and the measured states $[\alpha \ q \ \theta]^T$, respectively. The complete system is henceforth denoted $G_{\text{A/C}}$. This system is used when synthesising the offline controllers and when analysing the stability margins of the LQ controller.

The deflection intervals of the control surfaces are $\{-50^\circ, 25^\circ\}$ for the canard, $\delta_c$, and $\{-28.72^\circ, 28.72^\circ\}$ for the elevator, $\delta_e$. Both control surface types have a rate limit of $56^\circ$/s. The actuator dynamics are the same for both control surfaces, with a time constant $\tau = 1/20$. 

![Figure 2.3: Block diagram of the controlled system.](image)
In this chapter a theoretical description of the control strategies of interest is presented.

### 3.1 Linear quadratic control

In linear quadratic control the goal is to control a linear system by applying a control signal that minimises a quadratic cost function over an infinite time horizon [8, p. 267]. The problem can be formulated as

$$\begin{align*}
\min_u & \quad J = \int_0^\infty (x^T(t)Qx(t) + u^T(t)Ru(t)) \, dt \\
\text{s.t.} & \quad \dot{x} = Ax + Bu \\
& \quad y = Cx + Du
\end{align*}$$

(3.1)

where $Q$ and $R$ are penalty matrices for the states $x$ and the control signals $u$, respectively. The optimal control can be calculated by first solving the algebraic Riccati equation for $S$

$$A^T S + S A - S B R^{-1} B^T S + Q = 0,$$

(3.2)

and then calculating the optimal feedback gain matrix $L$ as

$$L = R^{-1} B^T S.$$  

(3.3)

The optimal control is now given as

$$u = -Lx.$$  

(3.4)
3.1.1 Stability

Since the controlled system is a Multiple Input Multiple Output (MIMO) system, stability properties of the LQ controller were analysed using disk margins. The disk margin provides an ellipse in a gain and phase margin plane within which the closed loop system is guaranteed to be stable. For MIMO systems the multiloop disk margins are particularly interesting. These describe the allowed variations in gain or phase for all input channels combined for the system to remain stable. The following theory is based on [2]. In Figure 3.1 the disk margins are presented in relation to the classical margins.

The multiloop disk margins are produced by breaking the loop in each input channel and introducing a perturbation gain as \((1 + \Delta)/(1 - \Delta)\), where \(\Delta\) is a complex number satisfying \(|\Delta| < d\) for some \(d\). The system is perturbed for each frequency until a maximum value of \(d\), that the stability of the system is guaranteed for, is found. Maximum gain and phase margin variations of the input channels can then be calculated as

\[
d g_m = \pm 20 \log_{10} \frac{1 + d}{1 - d} \tag{3.5}
\]

\[
d \phi_m = \left(\frac{180}{\pi}\right) 2 \arctan d. \tag{3.6}
\]

This way of analysing the closed loop system stability is more conservative than looking at the classical gain and phase margins and captures common modes of the MIMO system. For the sake of comparison the classical margins are also produced by breaking the loop at one input channel at a time. In both the disk margin and classical margin analysis the saturations are omitted.
3.2 Model predictive control

MPC is an attractive control strategy that directly accounts for constraints on states or control signals, since these are handled explicitly in the controller. The main idea is to formulate the control problem as an optimisation problem with an objective function and associated constraints. In each time step the optimisation problem is solved for a finite prediction horizon and the first input is realised. In Figure 3.2 the basic structure of MPC is shown [4].

![Figure 3.2: Basic structure of MPC [4, Fig. 1.2].](image)

### 3.2.1 Problem formulation

The following section is based on [5].

The discrete system discussed in this section is described by equations (3.7), where $z$ represents the states that are controlled.

\begin{align*}
    x(k+1) &= Ax(k) + Bu(k) \quad (3.7a) \\
    y(k) &= Cx(k) \quad (3.7b) \\
    z(k) &= Mx(k) \quad (3.7c)
\end{align*}

A common way of formulating an MPC problem is to use a quadratic cost function as in equation (3.8),

\[
    J_N(x(k)) = \sum_{j=0}^{N-1} \|z(k+j)\|_Q^2 + \|u(k+j)\|_R^2 
\]  

(3.8)

where $N$ is the prediction horizon and $Q \geq 0$, $R > 0$ are positive semi-definite and definite penalty matrices, respectively.
The reason why a quadratic cost function is commonly used is because the optimisation problem can then be solved with quadratic programming (QP). A typical formulation of a problem that is solvable with QP takes the form of equation (3.9).

\[
\begin{align*}
\min_w & \quad \frac{1}{2} w^T H w + f^T w \\
\text{s.t.} & \quad \xi w \leq b
\end{align*}
\]

(3.9)

\[\text{3.2.2 Compact description}\]

A compact description of the problem, \(N\) steps forward in time, can be formulated by vectorising the variables in order to make calculations and notation more manageable. The vectorised dynamics are introduced in equation (3.10),

\[X = Ax(k) + BU \]

(3.10)

where

\[
\begin{align*}
X &= \begin{bmatrix} x(k) \\ x(k+1) \\ \vdots \\ x(k+N-1) \end{bmatrix}, \\
U &= \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+N-1) \end{bmatrix},
\end{align*}
\]

(3.11)

\[
A = \begin{bmatrix} I \\ A \\ \vdots \\ A^{N-1} \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 & \ldots & 0 \\ 0 & 0 & \ldots & \ldots & 0 \\ \vdots & AB & B & \ldots & 0 \\ A^{N-2}B & \ldots & AB & B & 0 \end{bmatrix}.
\]

Let \(Q\), \(R\) and \(M\) be block diagonal matrices each with \(N\) blocks. Equation (3.8) is rewritten to a QP-problem as in equation (3.12), where \(F_x, f_x\) describe constraints on states and \(F_u, f_u\) on control signals. Note that \(\xi_1 \leq \xi_2\), where \(\xi_1\) and \(\xi_2\) both are vectors, is an element-wise inequality.

\[
\begin{align*}
\min_U & \quad (Ax(k) + BU)^T M^T QM (Ax(k) + BU) + U^T RU \\
\text{s.t.} & \quad F_x X \leq f_x \\
& \quad F_u U \leq f_u
\end{align*}
\]

(3.12)

After expanding the quadratic terms it is evident that some terms are constant and can be disregarded since they will not affect the optimisation. After further simplification, equation (3.12) can be rewritten as equation (3.13).

\[
\begin{align*}
\min_U & \quad \frac{1}{2} U^T (B^T M^T QMB + R) U + (B^T M^T QM A x(k))^T U \\
\text{s.t.} & \quad F_x X \leq f_x \\
& \quad F_u U \leq f_u
\end{align*}
\]

(3.13)
3.2.3 Reference tracking

In order to make reference tracking possible, the objective function needs to be extended as in equation (3.14). Applying the weighted norm of the error between measured output and reference signal ensures that the states will be controlled towards the reference.

\[ J_N(x(k)) = \sum_{j=0}^{N-1} \|z(k+j) - r(k+j)\|_Q^2 + \|u(k+j)\|_R^2 \]  

(3.14)

The reference signals, \( r(k) \), are vectorised over \( N \) time steps as in equation (3.15),

\[ \bar{R} = \begin{bmatrix} r(k) \\ r(k+1) \\ \vdots \\ r(k+N-1) \end{bmatrix} \]  

(3.15)

Using this \( \bar{R} \) it is possible to rewrite the objective function as in equation (3.16), which can be simplified in a similar fashion to equation (3.12), although this is omitted in this thesis.

\[
\min_U \left( M (Ax(k) + BU) - \bar{R} \right)^T Q \left( M (Ax(k) + BU) - \bar{R} \right) + U^T RU \\
\text{s.t.} \quad F_x X \leq f_x \\
\quad F_u U \leq f_u
\]  

(3.16)

3.2.4 Integral action

By augmenting the system with a constant disturbance \( d_k \) as in equation (3.17), integral action can be added to the controller.

\[
x(k+1) = Ax(k) + Bu(k) + Ed(k) \\
d(k+1) = d(k) \\
y(k) = Cx(k) + Du(k)
\]  

(3.17a, b, c)

The steady state reference, \( r \), is calculated by solving equation (3.18) where \( \hat{d}(k) \) is the estimated disturbance from a Kalman observer [14] and \( [x_r \quad u_r]^T \) denote the states and control signals in steady state.

\[
\begin{bmatrix} A-I & B \\ C & D \end{bmatrix} \begin{bmatrix} x_r \\ u_r \end{bmatrix} = \begin{bmatrix} -E\hat{d}(k) \\ r \end{bmatrix}
\]  

(3.18)

A common way to model the noise in a system is to introduce a process noise \( w(k) \) and measurement noise \( v(k) \), as in equation (3.19). Both are white noises
with intensities $R_1$ and $R_2$, respectively. The cross-covariance between $w(k)$ and $v(k)$ is denoted $R_{12}$.

$$x(k+1) = Ax(k) + Bu(k) + Nw(k)$$  \hspace{1cm} (3.19a)

$$y(k) = Cx(k) + Du(k) + v(k),$$  \hspace{1cm} (3.19b)

The prediction error $\hat{x} = x(k) - \hat{x}(k)$ is minimised with

$$\hat{x}(k+1) = A\hat{x}(k) + Bu(k) + K(y(k) - C\hat{x}(k) - Du(k)),$$  \hspace{1cm} (3.20)

where the Kalman gain $K$ is calculated by solving

$$K = (APCT + NR_{12})(CPC^T + R_2)^{-1}$$  \hspace{1cm} (3.21)

where $P$ is the positive semi-definite solution to the discrete Riccati equation [8, p. 146-149].

### 3.3 Fast MPC

A limitation of MPC is that online optimisation is computationally expensive and slow which makes it non-feasible for systems with fast dynamics. In this section a strategy for speeding up the optimisation is presented, based on [16].

Given the objective function in equation (3.23), the optimisation problem can be formulated as follows,

$$\min_{u} \quad l_f(x(k+N)) + \sum_{\tau=t}^{k+N-1} l(x(\tau), u(\tau))$$

s.t. \hspace{1cm} $F_f x(k+N) \leq f_f$,

$F_x x(\tau) + F_u u(\tau) \leq f$,

$x(\tau+1) = Ax(\tau) + Bu(\tau) + \bar{w}$,

$\tau = k, ..., k+N-1$

where $q$ and $r$ are weighting parameters, $l_f$ is a final cost function, $F_f$ and $f_f$ handle terminal constraints, $F_x$ and $F_u$ together with $f$ handle constraints on states and control signals, respectively, $A$ and $B$ describe the system dynamics and $\bar{w}$ is the mean of the process noise. The objective and final cost function are assumed to be quadratic and the constraints are assumed linear and weight matrices $Q \geq 0$ and $R > 0$. 

$$l(x(k), u(k)) = \begin{bmatrix} x(k) \\ u(k) \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} + q^T x(k) + r^T u(k)$$  \hspace{1cm} (3.23)
3.3 Fast MPC

3.3.1 Primal barrier interior-point method

A primal barrier interior-point method is formulated from equation (3.22) by first introducing the overall optimisation variable, with \( u \) and \( x \) defined as in Section 2.2,

\[
z = \begin{bmatrix} u(k)^T & x(k+1)^T & \ldots & u(k+N-1)^T & x(k+N)^T \end{bmatrix}^T
\]

and then rewriting the problem on a compact form

\[
\min_{z} \quad z^T Hz + g^T z \\
\text{s.t.} \quad Pz \leq h, \; Cz = b
\]

where

\[
H = \begin{bmatrix}
R & 0 & 0 & \ldots & 0 & 0 & 0 \\
0 & Q & S & \ldots & 0 & 0 & 0 \\
0 & S^T & R & \ldots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \ldots & Q & S & 0 \\
0 & 0 & 0 & \ldots & S^T & R & 0 \\
0 & 0 & 0 & \ldots & 0 & 0 & Q_f \\
\end{bmatrix}
\]

\[
P = \begin{bmatrix}
F_u & 0 & 0 & \ldots & 0 & 0 & 0 \\
0 & F_x & F_u & \ldots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \ldots & F_x & F_u & 0 \\
0 & 0 & 0 & \ldots & 0 & 0 & F_f \\
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
-B & I & 0 & 0 & \ldots & 0 & 0 & 0 \\
0 & -A & -B & I & \ldots & 0 & 0 & 0 \\
0 & 0 & 0 & -A & \ldots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \ldots & I & 0 & 0 \\
0 & 0 & 0 & 0 & \ldots & -A & -B & I \\
\end{bmatrix}
\]

\[
g = \begin{bmatrix}
q + 2S^T x(k) \\
q \\
r \\
q \\
r \\
q_f \\
\end{bmatrix}, \quad h = \begin{bmatrix}
f - F_x x(k) \\
f \\
\vdots \\
f \\
\vdots \\
f_f \\
\end{bmatrix}, \quad b = \begin{bmatrix}
A x(k) + \tilde{w} \\
\tilde{w} \\
\tilde{w} \\
\end{bmatrix}
\]

The term \( Q_f \) is a weight parameter for the final state and \( q_f \) is a bias term for the final state.
A logarithmic barrier term $\kappa \phi(z)$ is added to the objective function where $\kappa$ is a constant design parameter and

$$
\phi(z) = \begin{cases} 
\sum_{i=1}^{K} - \log(h_i - p_i z), & Pz \leq h \\
+\infty, & Pz > h 
\end{cases}
$$

(3.27)

where $h_i$ and $p_i$ are the $i$:th rows of $h$ and $P$, respectively, and $K$ denotes the number of rows in said matrices. The barrier term adds a large cost to the objective if the inequality constraints are near the outskirts of the feasible region, which acts as an approximation to the inequality constraints. Equation (3.25) is rewritten as,

$$
\min_z \ z^T Hz + g^T z + \kappa \phi(z) \\
\text{s.t.} \ Cz = b.
$$

(3.28)

The reason for introducing this barrier function is so that an infeasible start Newton method can be used to solve the optimisation problem.

### 3.3.2 Infeasible start Newton method

Infeasible start implies that the initial value $z_0$ does not need to satisfy the equality constraint, $Cz = b$, although it must satisfy the inequality constraint $Pz \leq h$. A dual variable $\nu$ is associated with the equality constraint and an optimality condition is formed with the primal and dual residual as in equation (3.29), where $\nabla \phi$ denotes the gradient of $\phi$.

$$
\begin{align*}
  r_d(z, \nu) &= 2Hz + g + \kappa \nabla \phi(z) + C^T \nu = 0 \\
  r_p(z) &= Cz - b = 0
\end{align*}
$$

(3.29)

Starting at $z = z_0$, at each iteration the norm of the residuals is compared to a small value $\epsilon$ and if the norm is below the threshold, an approximate optimal solution has been found for that iteration. If the norm is larger than $\epsilon$, a Newton step is made to decrease the norm of the residuals. To achieve this, a search direction for the primal and dual variable is found by solving equation (3.30), where $\nabla^2 \phi$ denotes the Hessian of $\phi$.

$$
\begin{bmatrix}
  2H + \kappa \nabla^2 \phi(z) & C^T \\
  C & 0
\end{bmatrix} 
\begin{bmatrix}
  \Delta z \\
  \Delta \nu
\end{bmatrix} = 
\begin{bmatrix}
  -r_d \\
  -r_p
\end{bmatrix}
$$

(3.30)

When the search direction has been established, a step length $s \in (0, 1]$ is calculated with a backtracking line search which decreases the norm of the residuals while still satisfying $Pz \leq h$. The norm of the residuals is defined as in equation (3.31).

$$
\begin{align*}
f(z, \nu) &:= ||r_d(z, \nu), r_p(z)||_2
\end{align*}
$$

(3.31)
3.3 Fast MPC

The variables $z$ and $\nu$ are updated between iterations and the process is repeated until the norm is sufficiently small, see Algorithm 3.1 for a description on how one iteration of the backtracking line search can be performed [3, p. 464].

**Algorithm 3.1:** Backtracking line search

1. Given $s \in (0, 1]$, $\alpha_l \in (0, 0.5)$, $\beta_l \in (0, 1)$, $\Delta z$, $\Delta \nu$;
2. $z^+ = z + s \Delta z$;
3. $\nu^+ = \nu + s \Delta \nu$;
4. while $f(z^+, \nu^+) > (1 - \alpha_l s) f(z, \nu)$ do
   5. $s = \beta_l s$;
   6. $z^+ = z + s \Delta z$;
   7. $\nu^+ = \nu + s \Delta \nu$;
5. end

3.3.3 Further approximations

Another approximation is warm starting the algorithm with the optimisation variable $z$ initialised as a time-shifted version of the solution from the previous iteration, as in equation (3.32). For notational simplicity, $z_k := z(k)$, $u_k := u(k)$, $x_k := x(k)$, while $N$ is the prediction horizon of the controller.

\[
\begin{align*}
    z_{k-1} &= \begin{bmatrix} u_{k-1} & x_k & \cdots & u_{k+N-2} & x_{k+N-1} \end{bmatrix}^T \\
    z_{k,\text{init}} &= \begin{bmatrix} u_k & x_{k+1} & \cdots & u_{k+N-2} & x_{k+N-1} & u_{k+N-2} & x_{k+N-1} \end{bmatrix}^T
\end{align*}
\]  

(3.32)  

(3.33)

The last control signal and predicted state, $u_{k+N-2}$ and $x_{k+N-1}$, are repeated at the end of the initialisation vector to satisfy the rate constraints for the control signal. Warm starting makes the backtracking line search converge in fewer steps because the solution at time $k$ will not differ much from the solution at time step $k - 1$ since the sampling rate is sufficiently high.

Calculating the exact optimal solution to the control problem is not necessary to achieve good results, an approximation that is close enough to the true optimum is adequate. By setting a fixed iteration limit for the algorithm described in Section 3.3.2 the computational time of the controller can be reduced. The fixed iteration limit is chosen by comparing simulations while decreasing the iteration limit between runs. The iteration limit can be decreased with unchanged results, as long as the approximate solution does not differ too much from the true optimum.

To sum up the Fast MPC approach, pseudo code for how the controller solves one iteration is found in Algorithm 3.2, where $\delta$ and $\epsilon$ are some small positive constants and $K_{\text{max}}$ is the maximum allowed number of iterations.
Algorithm 3.2: Fast MPC iteration, with approximations

1. $K = 0$;
2. while not $(f(z, v) < \varepsilon$ and $|Cz - b| < \delta$) and $K < K_{\text{max}}$ do
3. Warm start $z = z_{\text{init}}$;
4. Calculate step direction $\Delta z$ and $\Delta v$;
5. Take Newton step with backtracking line search;
6. Update $z$ and $v$;
7. $K = K + 1$;
8. end
The two control strategies were implemented for different flight situations and sets of constraints. The performance of the implemented controllers were evaluated using MATLAB and ARES during offline development.

A mutual element in the design process for both controllers is tuning the weight matrices. Adding a large weight to $\theta$ was central in achieving DLC behaviour in both implementations since it counteracts changes in pitch.

### 4.1 LQ controller

A relatively simple LQ controller was implemented using the theory from Section 3.1 and model (2.5). The controller was extended to meet desired properties and increase the performance.

#### 4.1.1 Baseline controller

To get accurate reference tracking with the LQ controller, the model used to derive the controller was expanded with integrator states $i_\theta$ and $i_\gamma$. The expanded model follows:

$$
\begin{bmatrix}
\dot{\alpha} \\
\dot{q} \\
\dot{\theta} \\
i_\theta \\
i_\gamma
\end{bmatrix} =
\begin{bmatrix}
A & 0 & 0 \\
0 & -1 & 0 & 0 \\
0 & 1 & -1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\alpha \\
q \\
\theta \\
i_\theta \\
i_\gamma
\end{bmatrix} +
\begin{bmatrix}
B & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
r_\theta & 0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\delta_c \\
\delta_e \\
r_\theta \\
r_\gamma
\end{bmatrix},
$$

(4.1)
where $r_\theta$ and $r_\gamma$ are reference signals for $\theta$ and $\gamma$, respectively. This gives the optimal LQ control as

$$\frac{u}{u_i} = -\begin{bmatrix} L | L_i \end{bmatrix} \begin{bmatrix} \alpha \\ q \\ \theta \\ i_\theta \\ i_\gamma \end{bmatrix}. \quad (4.2)$$

By introducing the error as

$$e_\theta = r_\theta - \theta$$
$$e_\gamma = r_\gamma - \gamma \quad (4.3)$$

and integrating to get the integrator states

$$i_\theta = \int e_\theta$$
$$i_\gamma = \int e_\gamma \quad (4.4)$$

reference tracking could be achieved through integral action. A block diagram showing how the states are fed back can be seen in Figure 4.1.

**Figure 4.1: Block diagram of the state feedback.**

The penalty matrices described in Section 3.1 are chosen so that a relatively high penalty is put on both integrator states and some penalty is put on the $\theta$ state.

#### 4.1.2 Feedforward of reference

The control scheme as described so far only drives the system to the reference via integral action. To speed it up a feedforward of the reference signal was intro-
duced. The control becomes
\[ \bar{u} = -L(x + R_f r) - L_i x_i \]  \hfill (4.5)
where
\[ x = \begin{bmatrix} \alpha \\ q \\ \theta \end{bmatrix}, \quad x_i = \begin{bmatrix} i_\theta \\ i_\gamma \end{bmatrix}, \quad r = \begin{bmatrix} r_\alpha \\ r_q \\ r_\theta \end{bmatrix} \]  \hfill (4.6)
with \( r_\alpha = r_\theta - r_\gamma \) and \( r_q = 0 \). The reference feedforward matrix \( R_f \) is
\[ R_f = a \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \]  \hfill (4.7)
where \( a \) is a positive number less than 1. This \( a \) is chosen to get a desired aggressiveness in the reference tracking. The second column of \( R_f \) can be set to 0 as \( q \) is not controlled. The result of how the feedforward affects the system can be seen in Figure 5.27 (Appendix 5.C).

### 4.1.3 Anti-windup compensator

When introducing control signal saturations the integrator states will windup once the control signals get saturated. This happens because the controller tries to force the system to reach the reference by increasing the integral states \( i_\theta \) and \( i_\gamma \).

There are several ways to overcome this problem. One way is to try to keep the unsaturated commanded control signal amplitude and rate as close as possible to those of the saturated control signal [15]. This can be done via a feedback of the difference between the saturated and the unsaturated control signals. When the control signal is saturated the difference is nonzero. This difference is multiplied with a gain matrix \( K_{aw} \), integrated together with the error and fed to the integrator feedback controller matrix \( L_i \). The anti-windup input of \( L_i \) can be chosen as an identity matrix to get full influence over the outputs. The gain matrix \( K_{aw} \) is chosen so that a saturated canard mostly affects \( i_\theta \) and a saturated elevator mostly affects \( i_\gamma \) since the elevator is mainly used to generate changes in \( \gamma \) and the canard is mainly used to cancel the generated moment. Naturally, the canard has some impact on \( \gamma \) and the elevator on \( \theta \) and thus a small contribution must be given to \( i_\gamma \) and \( i_\theta \) from a saturated canard and elevator, respectively. The expanded controller \( \bar{L}_i \) becomes
\[ \bar{L}_i = \begin{bmatrix} L_i & 1 \\ 0 & 1 \end{bmatrix}. \]  \hfill (4.8)

The control signal contribution from the integrator feedback becomes
\[ u_i = -\bar{L}_i \int \left[ K_{aw}(\bar{u} - \bar{u}_{sat}) \right]. \]  \hfill (4.9)
The full system block diagram with both the anti-windup and the reference feed-forward can be seen in Figure 4.2 and the difference in commanded control signals with and without anti-windup can be seen in Figures 5.28 and 5.29 (Appendix 5.C).

**Figure 4.2**: Full system block diagram. Note that the control signal saturations are normally included in $G_{A/C}$ but have been extracted for illustrative purposes.

### 4.2 Fast MPC

The Fast MPC strategy was implemented in MATLAB in order to simulate its performance and computational speed. The formulation of the control problem was done according to the theory presented in Section 3.3 and the model described in Section 2.2.

### 4.2.1 Matrix definitions

The control problem at hand has an objective function and constraints that are separable in state and control, which means that $S = 0$ in equation (3.23) [16]. Furthermore, the objective function is constructed with $H$ as in Section 3.3, with the exception that $S = 0$ and $g = -2z_r^T H$ where $z_r = [x_r^T, u_r^T]^T$. This follows from when reference tracking is added to the objective function,

$$
(z - z_r)^T H(z - z_r) = z^T H z - 2z_r^T H z + z_r^T H z_r
$$

and the constant term $z_r^T H z_r$ is removed since it will not affect the minimisation. Equation (4.10) is compared with (3.25) and $g$ can be identified as above [1]. The reference vector $z_r$ is found by solving the system of equations (4.11). The vector $r$ is the reference signal and $x_r$ and $u_r$ are the references for the states and control signals in steady state, respectively [16].
4.2 Fast MPC

The matrix $P$ is implemented as in Section 3.3 and $Q_f$ is chosen as the positive definite solution to the discrete algebraic Riccati equation used in discrete LQ-problems, and acts as a terminal penalty on the last state. This approach assumes that an LQ controller can be run from $k = N$ to $k = \infty$ [1].

No explicit terminal constraints are put on the states, thus $F_x = 0$ and $f_f = 0$. Additionally, the mean process noise $\bar{w}$ is considered a design parameter when implementing integral action for the controller. The matrices $C, b$ and $h$ are constructed as described in Section 3.3.

4.2.2 Constraints

Constraints are imposed on both the amplitude and rate of the control signals. No constraints are imposed on the states. The rate constraints between samples are described by equation (4.12a), minimum and maximum amplitude constraints are described by equation (4.12b) and (4.12c) and rate constraints respective to the control signal implemented in the previous iteration are described by equations (4.12d) and (4.12e) with $u_k := u(k)$.

\[
\begin{align*}
|u_{k+1} - u_k| &\leq \Delta u \\
-u_k &\leq u_{\text{max}} \\
u_k &\geq u_{\text{min}} \\
u_k &\leq \Delta u + u_{k-1} \\
-u_k &\leq \Delta u - u_{k-1} \\
k &= 1, \ldots, N - 1
\end{align*}
\]

(4.12a, 4.12b, 4.12c, 4.12d, 4.12e)

To achieve this, $F_x = 0$,

\begin{equation}
F_u = \begin{bmatrix}
u_{k+1} - u_k \\
-u_k + u_{k+1} \\
u_k \\
-u_k \\
u_k \\
-u_k
\end{bmatrix}, \quad f = \begin{bmatrix}
\Delta u \\
\Delta u \\
u_{\max} \\
u_{\min} \\
\Delta u + u_{k-1} \\
\Delta u - u_{k-1}
\end{bmatrix}
\end{equation}

(4.13)

where $\Delta u$ corresponds to the maximum allowed rate of the control signal. Since $P$ is block diagonal, see equation (3.26), the rate constraint respective to the control signal implemented in the previous iteration will be applied to all future control signals if $f$ in equation (4.13) is used. To resolve this issue, equation (4.13) is used only in the first iteration and for all other iterations another matrix $f_2$ is introduced as in equation (4.14), where $a \geq u_{\text{max}} + \Delta u$. 

\[ f_2 = \begin{bmatrix} \Delta u_{\text{max}} \\ \Delta u_{\text{min}} \\ u_{\text{max}} \\ u_{\text{min}} \\ a \\ a \end{bmatrix} \] (4.14)

### 4.2.3 Integral action

A process noise \( w \) with constant mean \( \bar{w} \) is introduced to the system and estimated with a Kalman observer in order to add integral action to the controller as described in Section 3.2.4. The estimated states and disturbances are expressed as the vector \( \hat{x}_{\text{est}} \) in equation (4.15),

\[ \hat{x}_{\text{est}} = \begin{bmatrix} \hat{x} \\ \hat{d} \end{bmatrix} \] (4.15)

where

\[ \dot{x} = \begin{bmatrix} \dot{\hat{\alpha}} \\ \dot{\hat{q}} \\ \dot{\hat{\theta}} \end{bmatrix}, \quad \dot{d} = \begin{bmatrix} \dot{\hat{d}}_1 \\ \dot{\hat{d}}_2 \\ \dot{\hat{d}}_3 \end{bmatrix} \] (4.16)

The Kalman filter gain \( K \) is calculated using the built-in MATLAB function \( \text{kalman}(\text{sys}, Q_n, R_n, N_n) \). The system \( \text{sys} \) is constructed as in equation (4.17a) and equation (4.17b) indicates that only the estimated states \( \hat{x}_k \) are measurable.

\[ \begin{bmatrix} \hat{x}_{k+1} \\ \hat{d}_{k+1} \end{bmatrix} = \begin{bmatrix} A & I \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \hat{d}_k \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u \] (4.17a)

\[ \dot{\gamma} = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \hat{d}_k \end{bmatrix} \] (4.17b)

The covariance matrices \( Q_n = R_1 \) and \( R_n = R_2 \) are chosen as diagonal matrices and the weights are determined using trial and error. The cross-covariance matrix \( N_n = R_{12} \) is chosen to be zero. By updating the estimation each time step according to equation (4.18) and calculating the steady state reference \( r \) and Kalman gain \( K \) as described in Section 3.2.4, integral action is added to the controller.

\[ \begin{bmatrix} \hat{x}_{k+1} \\ \hat{d}_{k+1} \end{bmatrix} = \begin{bmatrix} A & I \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{x}_k \\ \hat{d}_k \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + K (y - C \hat{x}_k) \] (4.18)

The process noise mean \( \tilde{w} = [\tilde{w}_1 \quad \tilde{w}_2 \quad \tilde{w}_3]^T \), also described in Section 3.2.4, is determined using trial and error while tuning the covariance matrices \( Q_n \) and \( R_n \).
4.2.4 Tuning parameters

The prediction horizon, $N$, is limited by the computational time needed to solve the optimisation in each time step. In Figure 5.30 and 5.31 (Appendix 5.D) the step responses with low and high prediction horizon are presented, respectively. With the low prediction horizon ($N = 2$) the system becomes oscillatory in the control signals and is also observed to be less robust to model errors. With the high prediction horizon ($N = 8$) the computational time is about four times larger than with $N = 5$, the value chosen for this implementation, without noticeable increase in performance.

Weight matrices of the baseline MPC controller are tuned by keeping $R$ constant and varying the diagonal elements of $Q$. The Kalman filter covariance matrices $Q_n$ and $R_n$ are tuned in a similar fashion with $R_n$ kept constant.

Tuning of the parameters related to Fast MPC is performed by first finding a stable tuning for the baseline controller, without the Kalman filter, and then varying the relevant Fast MPC parameters. For the infeasible start Newton method, $\kappa$ is decreased and $\epsilon$ is increased until the performance starts to degrade. The parameters related to the backtracking line search, $\alpha_l$ and $\beta_l$, are found through trial and error. The iteration limit $K_{max}$ is also decreased until the performance of the controller starts to degrade in order to produce a fast, yet still sufficiently precise controller.

4.3 Pilot interface

One of the aims of implementing direct lift is to give the pilot a more immediate control of the flight path angle, $\gamma$. By combining direct lift with automatic throttle control, as in this implementation, the pilot only needs to adjust the flight path angle to a desired value and then release the stick. The control system maintains the angle and the pilot only needs to stabilise the lateral dynamics. This will reduce the amount of pilot inputs during approach and landing [12].

4.3.1 Delta mode

During testing in high-fidelity flight simulation rigs it was decided that controlling the flight path angle with stick input $\Delta \gamma$ felt natural for making small adjustments during landing. Stick control was implemented so that a maximum forward deflection of the stick produces $\Delta \gamma = -3^\circ$ and maximum aft deflection produces $\Delta \gamma = 3^\circ$ in soft stop and $\Delta \gamma = 3.5^\circ$ in hard stop. Hard stop is an extra level of deflection in the aft direction which is used to give the pilot more control authority in extreme cases. The stick input values in between are linearly interpolated from neutral stick input which corresponds to $\Delta \gamma = 0^\circ$, see Figure 4.3. Maximum control authority for $\Delta \gamma$ is limited by how much lift can be produced with direct lift, which in turn is limited by the size of the control surfaces and the velocity of the aircraft.
4.3.2 Set-point mode

During approach the pilot should be able to control the flight path angle to a desired set-point value. With information from, for example, a control tower, the aircraft is controlled to a desired flight path angle by rotating $\theta$ and keeping $\alpha$ constant. When the pilot switches to delta mode the system automatically maintains the set flight path angle. In set-point mode, the stick input controls the rate of the flight path angle, $\dot{\gamma}$. Stick deflection angles are mapped to rates that give a satisfactory response and neutral stick corresponds to $\dot{\gamma} = 0$, see Figure 4.3.

![Figure 4.3: Stick input in the two different modes. The grey circles indicate different stick deflections. Note that hard stop is not used in set-point mode.](image-url)
4.4 Simulation setup

In Figure 4.4 a crude overview of the system used in ARES is presented.

![Figure 4.4: Overview of the simulation system.](image)

Data is sampled from the simulation environment with a frequency of 120 Hz. The relevant states and pilot inputs are fed to both the MPC and LQ controllers, which calculate control surface deflection angles for the canard and elevator. Note that the MPC controller runs at 30 Hz to reduce computational time. The switch block chooses which control signal to use depending on flags set by the user.

Alongside the direct lift controllers, two accessory controllers are run simultaneously at 120 Hz. The auto-throttle calculates throttle control input to maintain current air speed based on relevant states. The lateral controller calculates control surface deflection angles based on states and lateral pilot input. Both of the accessory controllers are supplied by Saab’s simulation environment and thus no development has been done for these controllers during the thesis.

4.4.1 Modifications for DLC

The simulation environment ARES is developed for a conventional controller and some modifications to the underlying structure were necessary when implementing DLC. During flight the aircraft is automatically trimmed so that if the pilot releases the stick, the aircraft control is trimmed to maintain the current flight path. This trim function changes the trim value for the angle of attack, $\alpha_{\text{trim}}$, which in turn degrades the performance of the direct lift controllers since the deviation from a trimmed state, $\Delta\alpha = \alpha - \alpha_{\text{trim}}$, is the value that is controlled.
The value of $\alpha_{\text{trim}}$ is acquired from the conventional control laws that run in parallel with the developed DLC control laws. When a large pilot input is fed to the controller, $\alpha_{\text{trim}}$ temporarily changes value which is unwanted when controlling with DLC as it disturbs the attitude of the aircraft. To mitigate this problem, a moving average smoothing filter is implemented which filters the last 2000 samples (16.66 seconds) of $\alpha_{\text{trim}}$ in order to remove the rapid changes in trim. How the filter affects $\alpha_{\text{trim}}$ is presented in Figure 4.5 where soft stop aft stick deflection is applied at time $t = 10$.

*Figure 4.5: Moving average smoothing filter for $\alpha$. Soft stop aft stick deflection is applied at $t = 10$.***
In this chapter the results produced during the thesis are presented. First, a comparison between the two direct lift controllers is shown. The DLC strategy is then compared against conventional flying with both simulations from ARES (Sections 5.1 and 5.2.1) and a landing study performed in a high-fidelity flight simulation rig (Section 5.2.2). Note that maximum aft stick deflection corresponds to a soft stop stick deflection in this chapter.

5.1 Direct lift control

In this section the implemented LQ and MPC controllers are compared with regards to performance, robustness and complexity. Several different measures were produced in ARES to evaluate all the aforementioned aspects. In order to keep the results clear and concise only a few figures are used as illustrative examples. The reader is referred to Appendix 5.A for additional plots related to the matters discussed in this section.

5.1.1 Delta mode

The following results are related to the behaviour obtained when a pilot input is fed to the controller in delta mode, see Section 4.3 for a detailed description on how the control modes work.

In Figure 5.1 the response from an instantaneous maximum aft stick deflection at time $t = 10$ is shown while in Figure 5.17 (Appendix 5.A) the response from a corresponding forward stick deflection is shown.
Figure 5.1: Step response from a maximum aft stick deflection at time $t = 10$ in delta mode.

In Table 5.1 the time constant and rise time for different stick deflections are presented for both controllers. The stick deflections are applied when the aircraft is in a trimmed state with $\gamma = 0$. The time constant $\tau$ is defined as the time it takes to reach 63% of the final value and the rise time $T_r$ is defined as the time between going from 10% to 90% of the final value.
5.1 Direct lift control

Table 5.1: Time constant $\tau$ and rise time for $\gamma$ with different stick inputs in delta mode. Positive difference corresponds to the MPC being faster than the LQ.

<table>
<thead>
<tr>
<th>Controller</th>
<th>$\tau$ [s]</th>
<th>$T_r$ [s]</th>
<th>$\tau$ [s]</th>
<th>$T_r$ [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>LQ</td>
<td>1.40</td>
<td>2.00</td>
<td>1.43</td>
<td>1.97</td>
</tr>
<tr>
<td>MPC</td>
<td>1.23</td>
<td>2.29</td>
<td>1.21</td>
<td>2.20</td>
</tr>
<tr>
<td>Difference</td>
<td>0.17</td>
<td>-0.29</td>
<td>0.22</td>
<td>-0.23</td>
</tr>
</tbody>
</table>

5.1.2 Set-point mode

The figures and tables in this section are related to the behaviour obtained when the pilot input is fed to the controller in set-point mode, see Section 4.3.

From a trimmed flight state, maximum stick deflection is applied for two seconds and then released. The behaviour of the states is presented in Figure 5.2 and Figure 5.18 (Appendix 5.A).

Figure 5.2: Response from a maximum aft stick deflection at time $t = 10$ in set-point mode.
Figure 5.2: Response from a maximum aft stick deflection at time $t = 10$ in set-point mode.

The overshoot and settling time are presented in Table 5.2 and the time constant and rise time are presented in Table 5.3. Overshoot $M$ is defined as how many percent greater the maximum value of $\gamma$ is than the reference value while the settling time $T_s$ is the time it takes for the maximum deviation of $\gamma$ to reach a steady state within a 5% interval of the reference value.

Table 5.2: Overshoot and settling time for $\gamma$ with different stick inputs in set-point mode. Positive difference corresponds to the MPC being faster than the LQ.

<table>
<thead>
<tr>
<th>Stick input</th>
<th>Maximum aft</th>
<th>Maximum forward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Controller</td>
<td>$M$ [%]</td>
<td>$T_s$ [s]</td>
</tr>
<tr>
<td>LQ</td>
<td>18.60</td>
<td>4.55</td>
</tr>
<tr>
<td>MPC</td>
<td>-</td>
<td>3.72</td>
</tr>
<tr>
<td>Difference</td>
<td>-</td>
<td>0.83</td>
</tr>
</tbody>
</table>

Table 5.3: Time constant and rise time for $\gamma$ with different stick inputs in set-point mode. Positive difference corresponds to the MPC being faster than the LQ.

<table>
<thead>
<tr>
<th>Stick input</th>
<th>Maximum aft</th>
<th>Maximum forward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Controller</td>
<td>$\tau$ [s]</td>
<td>$T_r$ [s]</td>
</tr>
<tr>
<td>LQ</td>
<td>1.58</td>
<td>1.03</td>
</tr>
<tr>
<td>MPC</td>
<td>2.28</td>
<td>2.42</td>
</tr>
<tr>
<td>Difference</td>
<td>-0.70</td>
<td>-1.39</td>
</tr>
</tbody>
</table>
5.1.3 Disturbance rejection

This section will present sensitivity measures that were used when comparing the controllers.

Turbulence

A stochastic wind disturbance, turbulence, was applied when flying in a trimmed state. The resulting behaviour of the states and control signals in very light and light turbulence is presented in Figure 5.3 and Figure 5.19 (Appendix 5.A), respectively.

Numerical measures such as the mean, $\mu$, and variance, $\sigma^2$, can be extracted from the data. The results are presented in Table 5.4 where the difference is calculated as the value for MPC subtracted from the value for LQ.
Table 5.4: Mean and variance of $\gamma$ for different levels of turbulence.

<table>
<thead>
<tr>
<th>Turbulence</th>
<th>Controller</th>
<th>$\mu$</th>
<th>$\sigma^2$</th>
<th>$\mu$</th>
<th>$\sigma^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Light</td>
<td>LQ</td>
<td>0.0050</td>
<td>0.0166</td>
<td>0.0118</td>
<td>0.1866</td>
</tr>
<tr>
<td></td>
<td>MPC</td>
<td>0.0312</td>
<td>0.0004</td>
<td>0.0249</td>
<td>0.0037</td>
</tr>
<tr>
<td>Difference</td>
<td></td>
<td>-0.0262</td>
<td>0.0162</td>
<td>-0.131</td>
<td>0.1829</td>
</tr>
<tr>
<td>Light</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Constant wind**

In Figure 5.4 the response from a change in horizontal wind speed from 0 m/s to 5 m/s is shown. The wind falls at 60° incidence in relation to true north in the simulation environment.

![Graphs showing response from a horizontal wind disturbance for LQ and MPC controllers.](image)
In Figure 5.5 the response from a change in vertical wind speed from 0 m/s to 3 m/s is shown, where a positive vertical wind is defined to flow straight downward in relation to the ground.

![Graphs showing response to vertical wind disturbance](image)

(a) LQ.

(b) MPC.

*Figure 5.5: Response from a vertical wind disturbance.*

### 5.1.4 Stability

The result of the disk margin stability analysis of the LQ controller described in Section 3.1.1 can be seen in Figure 5.6. For comparison, the Nyquist diagrams of the system at each input channel can be seen in Figure 5.7.
Figure 5.6: Disk margin of the LQ controller. Allowed gain variation up to ±5.35 dB and phase variation up to 33.26°. Stability is guaranteed for all variations in the input inside the ellipse.

Figure 5.7: Nyquist diagrams of the system at each input channel. Classical stability margins can be extracted for each channel.
5.2 Comparison against conventional control

The LQ and MPC controllers were compared with conventional flying by studying the properties of the path angle. A landing study in a high-fidelity flight simulation rig was also performed to determine if the touch down precision and pilot workload was improved.

5.2.1 Transient response

The transient response for a maximum aft stick deflection at time \( t = 10 \) can be seen in Figure 5.8 and the corresponding control surface response is presented in Figure 5.9. Note that both DLC controllers have a faster response compared to the conventional controller during the first 0.6 – 0.7 s.

![Figure 5.8: Transient response of maximum aft stick deflection of the two DLC controllers and conventional control.](image1)

![Figure 5.9: Control surface response of maximum aft stick deflection of, from left to right: MPC, LQ and conventional control.](image2)
5.2.2 Landing study

Three landings were conducted with each controller by an experienced Saab test pilot. The approach was initialised at 600 m at 95 m/s in a trimmed state, roughly 4 km from the runway. Only delta mode was used in the study. The pilot’s objective was to land the aircraft at a specified point on the runway while maintaining $\gamma = -3^\circ$ throughout the landing sequence. This is to simulate a scenario similar to an aircraft carrier landing. Light turbulence with a constant side wind of 3 m/s was applied in all test flights. Figures 5.10-5.12 shows the altitude and flight path angle for one of the test flights for each controller. Note that the altitude shown in the figures is the pressure altitude, which places the runway at an altitude of roughly 50 m. Plots for the remainder of the test flights are found in Figures 5.21-5.26 (Appendix 5.B). Figures 5.13-5.15 shows the trace of the stick deflections made by the pilot for all three test flights for each controller. The axis of the figures are chosen so that the whole stick deflection authority is shown, in order to illustrate that the stick deflection is close to neutral for the majority of the landing sequence. Figure 5.16 shows the touchdown spot for all test flights, with the desired touchdown spot marked with an X and with the runway centerline and width plotted for reference.

\[\text{Figure 5.10: Altitude and flight path angle } \gamma \text{ during the first LQ test flight.}\]
5.2 Comparison against conventional control

Figure 5.11: Altitude and flight path angle $\gamma$ during the first MPC test flight.

Figure 5.12: Altitude and flight path angle $\gamma$ during the first conventional test flight.
**Figure 5.13:** Stick deflection during the LQ test flights. From left to right: test flight 1, 2 and 3.

**Figure 5.14:** Stick deflection during the MPC test flights. From left to right: test flight 1, 2 and 3.

**Figure 5.15:** Stick deflection during the conventional test flights. From left to right: test flight 1, 2 and 3.
Figure 5.16: Touch down points for all test flights. The black X marks the desired touch down spot.
5.A Additional plots

Figure 5.17: Step response from a maximum forward stick deflection at time $t = 10$ in delta mode.
Figure 5.17: Step response from a maximum forward stick deflection at time \( t = 10 \) in delta mode.

Figure 5.18: Response from a maximum forward stick deflection at time \( t = 10 \) in set-point mode.
Figure 5.19: Steady state behaviour in light turbulence.
Figure 5.20: Step response from a maximum aft stick deflection at time $t = 10$ during slow flight in delta mode.
5.B  Landing study - additional plots

Figure 5.21: Altitude and flight path angle $\gamma$ during the second LQ test flight.
Figure 5.22: Altitude and flight path angle $\gamma$ during the second MPC test flight.

Figure 5.23: Altitude and flight path angle $\gamma$ during the second conventional test flight.
Figure 5.24: Altitude and flight path angle $\gamma$ during the third LQ test flight.

Figure 5.25: Altitude and flight path angle $\gamma$ during the third MPC test flight.
Figure 5.26: Altitude and flight path angle $\gamma$ during the third conventional test flight.

5.C Results from LQ extensions

Figure 5.27: Transient response with and without the feedforward extension of the LQ controller.
Figure 5.28: Commanded control signals before and after amplitude and rate saturations without anti-windup compensation in the LQ controller. A step that saturates the canards is applied at time $t = 10$ and removed at $t = 20$. Notice the delay due to integrator windup.

Figure 5.29: Commanded control signals before and after amplitude and rate saturations with anti-windup compensation in the LQ controller. Once again a step that saturates the canards is applied at time $t = 10$ and removed at $t = 20$. Now the unsaturated control signal follows the saturated.
5.D MPC prediction horizons

**Figure 5.30:** Step response with prediction horizon $N = 2$.

**Figure 5.31:** Step response with prediction horizon $N = 8$. 
In this chapter, conclusions related to the problem formulation will be drawn based on the results presented in Chapter 5. The feasibility and usefulness of DLC will be discussed, comparisons between LQ and MPC are presented as well as a discussion on the methodology and how the work could progress after the thesis.

6.1 Direct lift control

This section presents an answer to the questions regarding the feasibility of direct lift control and how it compares to conventional control.

6.1.1 Feasibility

Early in the thesis it was concluded that direct lift was feasible for an aircraft with properties similar to JAS 39 Gripen based on simple simulations of linearised longitudinal models. One of the criteria for DLC to be regarded as a feasible alternative to conventional landing was that a sufficiently large lift force could be generated while keeping the attitude constant. The amount of lift that can be generated limits the rate in set-point mode and the control authority in delta mode. Results from simulations in ARES and from flight tests in the high-fidelity flight simulation rigs showed convincing evidence that the implemented controllers’ performance did not suffer from these limitations. A control authority of ±3°, see Figure 5.1, and the response times, see Table 5.1, in delta mode felt more than adequate for making adjustments during landing. The speed of change in γ in set-point mode was acceptable but since the response did not match the pilot input perfectly, as can be seen in Figures 5.2 and 5.18, it felt somewhat inaccurate.
However, this inaccuracy did not cause any major degradation to the overall performance of the controllers.

### 6.1.2 Comparison against conventional control

One result that corroborates the claim that the pilot has a more immediate control of $\gamma$ is shown in Figure 5.8. From this figure it is evident that both the DLC controllers outperform the conventional controller during initial response since the non-minimum phase behaviour is eliminated. A fast transient response in $\gamma$ has appeared to be important from the pilot’s perspective. Small adjustments in the presence of non-minimum phase behaviour make it difficult to predict aircraft response which is critical when landing.

The implemented controllers display comparable performance to the conventional controller in the landing study. The data in Figures 5.10-5.12 and Figures 5.21-5.26 indicate that the MPC controller is better at keeping the flight path angle constant than the conventional controller. The LQ controller has a oscillatory behaviour in $\gamma$ but performs well in keeping $\gamma_{\text{mean}}$ close to the desired $\gamma = -3^\circ$. Not shown in the figures is the pitch angle $\theta$, which is experienced to be almost constant with the DLC controllers while the conventional controller changes the pitch attitude during landing.

Regarding the stick input, the data from the study (Figures 5.13-5.15) demonstrate larger amounts of pilot input for the DLC controllers compared to the conventional controller. This might be explained by the limited control authority used in delta mode which forces the pilot to use larger longitudinal stick inputs to achieve the same response as in the conventional case. Furthermore, the pilot flew in a manner that was described as “not chasing the velocity vector” which in the pilot’s experience led to fewer pilot corrections during landing. The velocity vector is an indication for where the aircraft is headed and it is shown to the pilot on the heads up display. A less experienced pilot might be tempted to make more corrections, especially in the LQ landings since the flight path angle fluctuates around a correct mean, thus creating the same behaviour in the velocity vector.

In order to draw any meaningful conclusions about the touch down precision from the landing study, a larger amount of landings would have to be conducted. Although, from Figure 5.16 a case can be made that the DLC controllers at least display comparable touch down precision to the conventional controller.

During the test flights the pilot expressed some general comments regarding the performance of the DLC controllers. The overall feeling when landing with DLC was described as “good/okay” and that “…the flight path angle is held well”. The amount of stick input used was described as “very little at higher altitude and a little more closer to the ground”. Concerns about the control authority and the speed of the controllers if DLC was to be used operationally were also raised by the pilot. In conclusion, the pilot described flying with DLC as “…flying with a pre-selected attitude” and “flying with DLC feels, in some manner automatic” and described the implementation as an interesting idea which needs to be developed further from a system safety perspective if ever to be used operationally.
6.2 Comparison of implemented controllers

Another major focus of the thesis was to compare the LQ and MPC approaches with regards to performance, robustness and complexity. The conclusions concerning these measures are presented in this section.

6.2.1 Performance

From Table 5.1 it is clear that the rise time is smaller for the MPC although in Figure 5.1 the LQ controller reaches steady state faster. When performing landings in the test rigs it was concluded that the two controllers behave very similarly and the slight differences in rise time and settling time are insignificant from a pilot’s perspective.

When pilot input is fed in delta mode, the control authority is set to $\pm 3^\circ$ and from Figure 5.1 it is clear that the LQ controller is more accurate at controlling $\gamma$ to the desired reference and therefore ensures that the authority is as large as promised. The deviation from a nominal $\theta$ between 20 and 30 seconds in the figure is caused by the moving average smoothing filter for $a_{\text{trim}}$, since the large amount of samples delays $a_{\text{trim}}$ from returning to its nominal value, which is evident in Figure 4.5. The integral action through estimated disturbances with a Kalman filter implemented in the MPC is not as accurate as applying integral action through integrator states, as in the LQ, and requires a lot of ad hoc tuning. The difference in accuracy can be seen in Figure 5.5.

Keeping the attitude constant while adjusting $\gamma$ is another measure that is illustrative when evaluating direct lift. In Figure 5.1 we see that when a pilot input is fed to the controller a small temporary change in $\theta$ occurs. It is desirable to minimise the amplitude of the change in pitch as it might obstruct the pilots view of the landing strip. At speeds used for the data showed in the figure, both controllers perform similarly, but since the MPC controller is more sensitive to changes in flight envelope, when the speed or altitude is changed the LQ controller performs better. This is presented in Figure 5.20 (Appendix 5.A).

6.2.2 Robustness and stability

The results in Figure 5.3 and Table 5.4 (as well as Figure 5.19) clearly show that both controllers are able to handle light turbulence, especially the MPC which has a significantly smaller variance in $\gamma$ compared to the LQ. The MPC controller has a larger mean compared to the LQ for both cases and this is once again related to the fact that the integral action of the LQ is more accurate than the one implemented in the MPC. The reason why only light and very light levels of turbulence are investigated is because when performing carrier landings out at sea there is mainly convective turbulence, which is almost negligible in normal weather conditions [9, p. 19-20].

For constant wind disturbances both controllers perform well, especially for horizontal disturbances where $\gamma$ is kept almost constant, see Figure 5.4. When applying a vertical wind disturbance, $\gamma$ deviates significantly from its reference,
see Figure 5.5. This result is natural since a vertical wind affects $\alpha$ more than a horizontal one because the vertical component of the velocity vector is relatively small. In both figures it is evident that $\theta$ drifts and takes a long time (roughly 30 seconds) to return to its nominal value after the disturbance. This is caused by the moving average smoothing filter for $\alpha_{\text{trim}}$ which reduces the amplitude of change but slows down the response as a consequence of filtering over a large amount of samples. It is apparent in Figure 5.5 that the MPC controller is significantly better at suppressing vertical wind disturbances. This is due to the Kalman filter being faster at compensating for constant errors than the LQ controller.

For the LQ controller a quick comparison of the multiloop disk margins presented in Figure 5.6 and the classical margins which can be roughly estimated from Figure 5.7, clearly indicates that the classical margins are overly optimistic as they appear to have infinite gain margins and very high phase margins. While it is true that the aircraft can still stabilise itself if either one of the input channels is completely cut off, it would only take a minor discrepancy in phase to then lose the stability. Due to the nonlinear nature of the MPC controller, no linear analysis of the stability margins could be done. Therefore, no guarantees could be made as to how stable it actually is. However, it should be noted that from comparative tests between the two controllers, performed in simulations, no major differences in stability have appeared.

### 6.2.3 Complexity

The intention of the complexity measure is to give a comparison of how complex the design process was as well as the computational complexity of the controllers.

In Table 6.1 and 6.2 the design parameters of the LQ and MPC are presented, respectively. It is obvious that the MPC controller has more design parameters and this was also reflected during the design process where more time had to be assigned to tuning the MPC controller than the LQ counterpart.

**Table 6.1: Design parameters for the LQ controller.**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Used in</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q$</td>
<td>Baseline controller</td>
<td>5x5 - diagonal</td>
</tr>
<tr>
<td>$R$</td>
<td>Baseline controller</td>
<td>2x2 - diagonal</td>
</tr>
<tr>
<td>$K_{aw}$</td>
<td>Anti-windup</td>
<td>2x2</td>
</tr>
<tr>
<td>$a$</td>
<td>Feedforward</td>
<td>constant</td>
</tr>
</tbody>
</table>
6.3 Practical observations

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Used in</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q$</td>
<td>Baseline controller</td>
<td>5x5 - diagonal</td>
</tr>
<tr>
<td>$R$</td>
<td>Baseline controller</td>
<td>2x2 - diagonal</td>
</tr>
<tr>
<td>$N$</td>
<td>Baseline controller</td>
<td>constant</td>
</tr>
<tr>
<td>$Q_n$</td>
<td>Kalman filter</td>
<td>6x6 - diagonal</td>
</tr>
<tr>
<td>$R_n$</td>
<td>Kalman filter</td>
<td>3x3 - diagonal</td>
</tr>
<tr>
<td>$\bar{w}$</td>
<td>Kalman filter</td>
<td>3x1</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>Fast MPC</td>
<td>constant</td>
</tr>
<tr>
<td>$\kappa$</td>
<td>Fast MPC</td>
<td>constant</td>
</tr>
<tr>
<td>$\alpha_l$</td>
<td>Fast MPC</td>
<td>constant</td>
</tr>
<tr>
<td>$\beta_l$</td>
<td>Fast MPC</td>
<td>constant</td>
</tr>
<tr>
<td>$K_{\text{max}}$</td>
<td>Fast MPC</td>
<td>constant</td>
</tr>
</tbody>
</table>

Computational complexity was an issue mainly addressed for the implementation of the MPC controller. As a consequence of needing a sufficiently long prediction horizon in the time domain while also keeping the computational time down, the MPC was designed to run at 30 Hz. This allowed for a smaller prediction horizon $N$ which reduced the computational complexity significantly but still retained the same prediction horizon in the time domain compared to running it at 120 Hz as the rest of the simulation system. The reduction of the sampling time was deemed not to degrade the overall performance of the controller and was necessary in order to make the controller run in the high-fidelity flight simulation rig where stricter requirements on the computational time of the controllers had to be fulfilled.

In the LQ case, the computational time needed to solve the matrix multiplications in each time step is negligible and does not need to be taken into consideration when designing the controller, since the optimal gain is calculated offline.

6.3 Practical observations

Most of the design was made on a model linearised in a flight case with a higher than desirable air speed at landing. This was mainly due to issues with an algorithm in ARES, used to find trimmed state given an air speed and an altitude, not being able to find a stable trim at low speeds. Since the controllers still performed well in landing scenarios when tuned using the higher speed linearised flight cases, no attempts were made to manipulate the function in ARES as it was deemed too time consuming and the implementation not being the main focus of the thesis.

Disturbance rejection was not a focus when synthesising the controllers and it is probable that the controllers could have been designed to have better disturbance suppression if more time would have been assigned to this task. The reason for not considering this in the design phase was that the focus of the thesis was to investigate the feasibility of DLC, not to implement controllers that are robust
in unfavourable conditions. Thus the results regarding robustness are to be seen as a consequence of prioritising a good response in $\gamma$, rather than a conscious design choice.

Something that was found to be more useful than expected during development was the testing in the high-fidelity flight simulation rig. A lot of issues with the controllers were, as previously mentioned, far from obvious when only looking at the plots from ARES but became evident when piloting the aircraft in the simulator. An important aspect when implementing DLC is to ensure that the response from pilot inputs feels natural, which is very difficult to achieve when developing in offline simulation environments.

### 6.4 Future work

Since the controllers were developed with no regards to disturbance suppression some further analysis of the sensitivity should be done. While the MPC controller is relatively unaffected by relevant disturbances as a consequence of the Kalman filter tuning, the disturbance attenuation of the LQ controller is not satisfactory when it comes to measurement disturbances. An approach to evaluate this aspect would be to look at the sensitivity functions. From Figure 5.1 and 5.5 it is evident that the LQ controller handles model uncertainties with no major problem while an external disturbance significantly affects the system. This might imply that the sensitivity function is small for relevant frequencies while the complementary sensitivity function is not.

Introducing gain scheduling in both controllers would make them less dependent on flight cases. For the MPC controller this is simply a matter of linearising more flight cases, extracting the system matrices from the model and feeding them to the controller as it already takes the matrices of its internal model as arguments. This is also true for the Kalman filter. The LQ controller would require that for each linearised model the controller gains are recalculated offline and then fed to the controller depending on flight case. For both controllers the penalty matrices would need to be changed in order to tune the controllers for each flight case.


