Atmospheric Disturbance Compensation in the VLTI Telescope

Diploma thesis

Fabian Schossau
September 2007

École Polytechnique Fédérale de Lausanne
Faculté STI Sciences et Techniques de l’Ingénieur
Laboratoire d’Automatique
Dr. Denis Gillet, MER
Philippe Mullhaupt, Ph.D.
Yvan Michellod

Darmstadt University of Technology
Mechatronics in Mechanical Engineering
Prof. Dr.-Ing. Rainer Nordmann
Atmospheric disturbance compensation in the VLTI Telescope

For improving the resolution of the Very Large Telescope Interferometer (VLTI) by rejecting the atmosphere disturbance on the spatial observations, a mechanical prototype is being developed at the EPFL.

The system is made of a double-stage structure. The first stage is a blade guiding structure driven by a high-accuracy nonlinear microstepper motor with a theoretical precision of 100 nm. The second stage mounted on the first one is a fine positioning piezo actuator that has theoretical nanometer accuracy. The overall structure is supposed to be able to follow a reference trajectory at very high accuracy (nanometer scale) over a stroke of 70 mm for frequencies up to 200 Hz.

In this context, it is proposed to develop and implement a LQG multivariable optimal controller, the performance of which will be compared to a decoupled dual-stage SISO design. As the prototype is available, it will be possible to measure, implement and test the algorithms in real conditions.

This project is proposed in collaboration with the Observatory of Geneva and with the European Southern Observatory in Munich.

Lausanne, le 17 avril 2007

Collaborateurs responsables:
M. Yvan MICHELLOD (tél. int. 3 25 64)
Dr. Philippe MUELLHAUPT (tél. int. 3 38 38)
Declaration

I hereby declare, that I have written this thesis without any help from others and without the use of sources other than those specified in the thesis itself.

This thesis, identical or similar in form, has not been available to any audit authority yet.

Lausanne, the 14th of September 2007

________________________
Fabian Schossau

In terms of free information exchange, I grant the Darmstadt University of Technology and the École Polytechnique Fédérale de Lausanne, the copyright of this thesis for educational and research purposes.

Lausanne, the 14th of September 2007

________________________
Fabian Schossau
Acknowledgements

I would like to thank the Laboratoire d’Automatique, especially Dr. Denis Gillet and Philippe Mullhaupt for the possibility to write my diploma thesis at the École Polytechnique Fédérale de Lausanne. I am grateful to Professor Nordmann for the supervision on behalf of the Darmstadt University of Technology and therefore the final realization of this project. I appreciate the help of the whole administrative staff in Darmstadt and Lausanne, as there are, the offices of the two institutes, the ERASMUS offices and the student centers.

My special appreciation goes to Yvan for the supervision of my thesis, many valuable discussions and the support throughout the project. Further I am deeply grateful to Eric for the fruitful discussions, remarks and support with MATLAB®. Most importantly I would like to acknowledge and thank Nadine and my parents for the encouragement, support and love that kept me going.
## Contents

**List of Figures**  
VI

**List of Tables**  
VII

**Table of abbreviations**  
VIII

1 Introduction  
1

2 Background  
3  
2.1 The VLTI and the purpose of PRIMA  
3  
2.2 The DDL design and the present control scheme  
5  
\hspace{1em} 2.2.1 Requirements  
5  
\hspace{1em} 2.2.2 Hardware  
6  
\hspace{1em} 2.2.3 Controller  
7  
2.3 Controlling fundamentals  
8  
\hspace{1em} 2.3.1 Kalman estimator  
8  
\hspace{1em} 2.3.2 Optimal control  
12  
\hspace{1em} 2.3.3 LQG  
14  
\hspace{1em} 2.3.4 Dual-stage tracking control  
16

3 Modeling the system  
18  
3.1 Coarse stage  
18  
\hspace{1em} 3.1.1 Permanent Magnet Stepper Motor  
18  
\hspace{1em} 3.1.2 Parameter identification  
19  
\hspace{1em} 3.1.3 Linearization  
22  
3.2 Fine stage  
23  
3.3 Overall model  
25  
3.4 System properties  
26

4 Multivariable Controller Design  
28  
4.1 Control scheme  
28  
4.2 Estimator design  
31  
4.3 Regulator design  
33  
4.4 Discretization  
36  
4.5 Conclusion and further remarks on the controller design  
38
### 5 Simulation and comparison to the SISO design

5.1 Closed loop properties ................................................. 40
5.2 Simulation of discrete controller ................................... 40
5.3 Comparison to SISO design ........................................... 44

### 6 Implementation

6.1 Hardware configuration ............................................... 46
6.2 Software implementation ............................................. 48
6.3 Results and discussion ............................................... 48

### 7 Conclusion

50
List of Figures

2.1 The optical layout of the VLTI with two telescopes \cite{14} ....................... 4
2.2 Basic scheme of PRIMA \cite{14} ...................................................... 5
2.3 Schematic representation of the DDL ............................................... 6
2.4 Present control scheme .................................................................. 7
2.5 Observer structure corresponding to (2.1) ....................................... 9
2.6 Extended Kalman Filter cycle of Prediction and Update ................. 12
2.7 LQG controlled plant ..................................................................... 15

3.1 PM-VR Hybrid Stepping Motor \cite{10} .............................................. 19
3.2 Identification of stepping motor parameters ................................... 21
3.3 Bode plot of the identified model ................................................. 24
3.4 Structure of model composition ..................................................... 25

4.1 Tracking controller ......................................................................... 28
4.2 Controller layout ........................................................................... 31
4.3 Varying the Kalman filters covariance matrix ................................. 32
4.4 Tuning \( Q \) according to step response ........................................ 34
4.5 Tuning \( Q \) according to Kolmogorov tracking ............................... 35
4.6 Tuning \( R \) parameters ................................................................ 36
4.7 Discrete control scheme .................................................................. 37

5.1 50 \( \mu \)m step response with saturating fine stage actuator ................ 41
5.2 5 \( \mu \)m step response ................................................................. 41
5.3 Impact of process and measurement noise on the tuned system ......... 42
5.4 Kolmogorov tracking plot ............................................................... 43
5.5 Fine stage action range during Kolmogorov-signal tracking ............. 43
5.6 Absolute error of Kolmogorov-signal tracking ............................... 44
5.7 Available response data for current control scheme performances .... 44

6.1 Hardware Setup .............................................................................. 46
6.2 Agilent Measurement Setup .......................................................... 47
6.3 The DDL in the setup environment .............................................. 47
6.4 Principal sequence of the implemented algorithm ......................... 48
List of Tables

3.1 Identified Parameters .................................................. 22
3.2 Eigenvalues of the model ............................................. 26
5.1 Available benchmark data ............................................. 45
5.2 Simulation results ...................................................... 45
### Table of abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A/D</td>
<td>Analog-to-digital converter</td>
</tr>
<tr>
<td>ARX</td>
<td>Parametric model of linear system, with an autoregressive (AR) part and an extra (X) input</td>
</tr>
<tr>
<td>AT</td>
<td>Auxiliary Telescope of the VLTI</td>
</tr>
<tr>
<td>D/A</td>
<td>Digital-to-analog converter</td>
</tr>
<tr>
<td>DDL</td>
<td>Differential Delay Line, a component of the PRIMA facility</td>
</tr>
<tr>
<td>DISO</td>
<td>Dual-input single-output systems</td>
</tr>
<tr>
<td>ESO</td>
<td>European Southern Observatory</td>
</tr>
<tr>
<td>HDD</td>
<td>Hard-Disk-Drive</td>
</tr>
<tr>
<td>LTR</td>
<td>Loop Transfer Recovery, a frequency shaping method in LQR design</td>
</tr>
<tr>
<td>LQG</td>
<td>Linear Quadratic Gaussian Regulator, combination of a LQR with a Kalman filter</td>
</tr>
<tr>
<td>LQR</td>
<td>Linear Quadratic Regulator</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple-input multiple-output systems</td>
</tr>
<tr>
<td>PM</td>
<td>Permanent magnet, as in PM-VR stepping motor</td>
</tr>
<tr>
<td>PRIMA</td>
<td>Phase referenced imaging and micro-arcsecond astrometric facility, an enhancement of the VLTI</td>
</tr>
<tr>
<td>OPD</td>
<td>Optical path difference, the difference between two received light beams originating from the same source, but received on two different locations.</td>
</tr>
<tr>
<td>RK4</td>
<td>Runge Kutta 4th order method</td>
</tr>
<tr>
<td>RMS</td>
<td>Root mean square, $\text{RMS} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2}$</td>
</tr>
<tr>
<td>SISO</td>
<td>Single-input single-output systems</td>
</tr>
<tr>
<td>UT</td>
<td>Unit telescope of the VLT</td>
</tr>
<tr>
<td>VLT</td>
<td>Very Large Telescope, a telescope on Cerro Paranal in Chile operated by the ESO</td>
</tr>
<tr>
<td>VLTI</td>
<td>Very Large Telescope Interferometer</td>
</tr>
<tr>
<td>VR</td>
<td>Variable reluctance, as in PM-VR stepping motor</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

The strive for extrasolar planetary research has become increasingly popular over the last years. A new astrometric measurement method, the microsecond-astrometry, promises further discoveries of characteristics of other planetary systems. Furthermore, the atmosphere blurs the observations taken from earth and more precise and accurate instruments are being developed to compensate these disturbances to earthbound observation.

The Differential Delay Line (DDL) developed by the EPFL and the Observatory of Geneva is one subsystem of an instrument to enable microsecond-astrometry in the Very Large Telescope Interferometer (VLTI). In addition, it is able to reduce the effect of atmospheric disturbance in other instruments and therefore improve the performance of earthbound observation with the VLTI.

In the last 15 years, due to increasing miniaturization, large-displacement nm-accuracy systems have become more attractive in the computer industry. The structure of the DDL is comparable to these structures: it requires a high-nanometer-accuracy but has to provide a full stroke over many millimeters. Therefore, the DDL comprises two stages, a coarse stage and a fine stage, which is further referred to as a dual-stage system.

In this diploma thesis, a multivariable optimal regulator is to be developed for this dual-stage system in the context of astrometric research within the VLTI. Therefore this thesis starts in section 2.1 with a short introduction to astrometry and the task of the DDL in this context. The necessary background information for the controller development is given in the following sections. Section 2.2 provides the requirements (Section 2.2.1) as well as the hardware layout (Section 2.2.2) to the design. Further the currently installed control scheme is reviewed in Section 2.2.3. The theoretical background for the multivariable controller with a state estimator, is provided in Section 2.3. It outlines the optimal controller and its principles of origin in Section 2.3.2 as well as the equations of the Kalman filter, which is the state estimator, in Section 2.3.1. The last two subsections of the theoretical background provide further information about the linear-quadratic gaussian control (Section 2.3.3) and the control approaches taken so far in the dual-stage tracking problem (2.3.4).

Chapter 3 describes the mathematical model of the DDL and the identification of the associated parameters. It is structured according to the special configuration of the overall system. In section 3.1, the model of the coarse stage is developed. Section 3.2
provides the model of the fine stage. The overall model is explained in Section 3.3 and analyzed in Section 3.4.

In Chapter 4, the multivariable controller and its components are developed, based on the model in Chapter 3. The layout of the controller is found in Section 4.1; it considers the processing of the requirements in the overall control structure. The design of the two components, state estimator and regulator, arranged in the previous section, is described in sections 4.2 and 4.3. This controller is to be transferred into the discrete time domain to be implemented on the real system. This process of discretization is the content of Section 4.4.

Chapter 5 contains the results of the simulation of the discrete controller. Hence a comparison to the control scheme previously installed on the system is realized. Furthermore, the simulated results are to be implemented on the real system. The hardware setup as well as the results of the implementation are found in Chapter 6. Chapter 7 provides a conclusion to the thesis with a short summary and a review of the most important results.
Chapter 2

Background

2.1 The VLTI and the purpose of PRIMA

The Very Large Telescope (VLT) is an European Southern Observatory (ESO) operated observatory on Cerro Paranal in Chile. It consists of four 8m-telescopes, the Unit Telescopes (UTs), and an optical interferometer, the VLTI. The VLTI allows to combine the four UTs and at least three of eight movable 1.8m-telescopes, the Auxiliary Telescopes (ATs) [14].

Astrometry measures the angular position of stars. The precision of this basic method depends on the aperture size. In optical interferometry the angular resolution is increased by combining the light from multiple telescopes. Herein, the resolution parameter equivalent to the aperture size is the maximum distance between the combined telescopes, the baseline.

When two light beams originate from the same source and are being picked up at two different positions on earth, they follow different path lengths. The combination of the received light beams lets them interfere. When they arrive in phase they superimpose and produce bright fringes (white-light-fringes), and when they arrive out of phase they cancel each other out.

The main aim of optical interferometry is to find the white-light-fringes of the observed object. They provide information about the distance, the angular size and the angular separation of the observed objects. According to the difference in the baseline, the rotation of the earth and atmospheric disturbance the beams’ paths are not of equal length. This difference in length has to be altered in order to produce the required interference pattern. Therefore, the optical path difference (OPD) between the two beams of light can be varied by internal delay lines to produce these fringes [28].

Figure 2.1 shows the light beams originating from the same source, travel through the focus of the telescopes, pass the delay lines and the final combination in the interferometry lab in order to produce interferometry fringes [14].

The VLTI contains many different subsystems, a short overview of these systems is given in [14]. The phase-referenced imaging and micro-arcsecond astrometric facility (PRIMA) expands the VLTI by the possibility to observe two stars simultaneously, a so called dual feed capability. This enhancement allows to observe fainter objects and
achieve higher resolutions on the other VLTI instruments. Furthermore PRIMA’s key objective is to expand the search for extra-solar planets and their birth environment. In addition to that, the high precision astrometry will increase the reach of the VLTI [12].

The basic principle of phase referenced imaging and micro-arcsecond astrometry with an interferometer is shown in figure 2.2. A bright star and a reference object are observed simultaneously in one telescope. One star is focused and therefore the fringe motion is to be the motion of the other object. The differential delay between these two beams of light has to be equalized to observe the fringe motion. This variation to the differential length of the two paths is simultaneous to the compensation of the OPD described before. As the angle between the two stars in one telescope is smaller than the angle between the two objects in different telescopes, the atmospheric disturbance has less impact on the wave front of the light and therefore a higher precision is achievable. For best performance the internal delay, the length of the baseline vector and the measurement of the fringe phase have to be known very precisely (nanometer-/micrometer-/$10^{-1}$ rad-accuracy) and the light paths throughout the system should be closely matched [19].

PRIMA requires four sub-systems to commit its tasks: star separators, DDLs, two fringe sensors(trackers) and a high accuracy metrology. This thesis provides a control algorithm for the DDLs, so only the DDL is briefly explained in the following paragraph. The other systems are well described in [11].

The star separator splits the light beams of one telescope in the beam of the faint
science object and the bright guide stars beam. These beams are fed through the large VLTI delay lines. Simultaneously to the main delay lines canceling the OPD, the DDL’s task is to alter the path length between the two beams to cancel out the differential delay.

A total differential optical path difference of up to 70 mm has to be compensated by the DDL within a 5 nm accuracy. PRIMA could be operated without the DDLs - then the delay would have to be compensated by the VLTI delay lines - but a much lesser resolution would be achieved as the bandwidth of the delay lines is a lot smaller than the bandwidth of the DDL and the positioning in the delay line not as accurate [12].

A further objective of the DDL is the atmospheric disturbance compensation. Usually the atmospheric disturbances have a frequency content of up to 500 Hz. This bandwidth is not to be compensated by the large delay lines of the VLTI, which are currently used to compensate these disturbances. Using the DDL’s high bandwidth will improve the compensation and therefore improve the performance of the other VLTI instruments.

2.2 The DDL design and the present control scheme

2.2.1 Requirements

According to the astronomic specifications the ESO defined system requirements. These are the basic guidelines for the mechatronic design.

The atmospheric disturbance can be described by a Kolmogorov signal. The Kolmogorov signal describes the perturbation to the wave front arriving at the telescope in terms of a stochastic model. This signal has a frequency content of up to 250 Hz, wherein small amplitudes (|x| < 1 µm) come along with high frequencies (f > 50 Hz) and large amplitudes with low frequencies.

The residual tracking error is to be under 70 nm root mean squared (RMS).

Further the system is operated under vacuum and no active cooling device is installed. Therefore the power dissipation of the plant is required to stay under 5 W.
In addition, according to the observation duration over a whole night, the system is in need of a full stroke of 70 mm. [26].

### 2.2.2 Hardware

In order to achieve the high nanometer accuracy a system separating the requirements of a coarse stroke and high nanometer accuracy was proposed. A piezoelectric actuator provides the nm-accuracy at a short stroke with a high bandwidth. And a coarse actuator provides the large stroke at a low bandwidth. So a combination of the actuators joins the advantages of each actuator and eliminates their disadvantages.

Altering the length of the OPD, in terms of large µm-scale- and very accurate nm-scale strokes, yields, that both actuators have to act on the same output, the length of the light path. This light beam is reflected through the optics into a cat’s eye which reflects the light further on its path. Changing the light beams length, by moving the cat’s eye or altering the path difference in the cat’s eye, varies the OPD.

The coarse actuator, a permanent-magnet (PM) stepper motor, moves, over a leadscrew, a blade-guiding structure on which the cat’s eye is mounted. The mirror in the center of the cat’s eye is attached to the top of a tripod piezoelectric and is able to be maneuvered at nanometer scale.

![Figure 2.3: Schematic representation of the DDL](image-url)
Figure 2.3 shows the simplified structure of the DDL. The light beam enters the cat’s eye on the top left, is reflected in the cat’s eye, passes the mirror mounted on top of the piezoelectric and leaves the cat’s eye on the bottom beam. One can easily see that changing the expansion of the fine stage or moving the whole blade-guiding-system by the action of the coarse stage alters the light beams path length.

In this thesis the elements guaranteeing the coarse movement, the blade-guiding-system, the permanent-magnet stepper-motor and the lead screw are considered as the coarse stage. As well as the element guaranteeing the fine movement of the system, the piezoelectric actuator, is referred to as the fine stage. A further description on the actuators is found in chapter 3.

2.2.3 Controller

The present control approach ([25],[26]) decouples the two stages into two single loops. Each loop takes the respective advantage of its actuator into account, and a partial observer is used to avoid the need for supplementary sensors.

The fine actuator control loop ensures fast positioning at a small range with high precision. The loop on the coarse actuator warrants the coarse positioning and avoids the saturation of the piezoelectric actuator.

To ensure an efficient tracking the microactuators loop is closed on the overall output of the system. This control scheme is illustrated in figure 2.4 and explained in the following paragraphs.

![Figure 2.4: Present control scheme, based on [25]](image)

**PM stepper motor control design** The stepper motor is not used in stepping mode, that means the motor coils are directly connected to voltage amplifiers.

To track a reference with the motor the controller generates an electrical field at the position on the stator with which the rotor shall align, the difference between the stator field and the equivalent reference field position is called $\phi$. To avoid the desynchronization between the stator and the rotor field, which would occur when the
stator field turns too fast for the rotor to follow, a saturation has been introduced on \( \frac{\partial \phi}{\partial t} \).

This system is stable as the rotor field will always try to align with the stator field and an integral-controller constantly summing up the position error is sufficient to drive the stepper to the desired position. A Butterworth-low-pass-filter has been implemented to treat only the low frequency content of the reference signal and to avoid the excitation of the steppers high-frequent resonant mode. The filters cut-off frequency is 10Hz.

**Piezo control design**  The fine stage controller is a proportional-integral controller. Its input is the error between the reference position and the measured position. A feed-forward term including the estimated coarse stage position is added to the controllers output voltage. The feedforward term on the input voltage of the plant improves the tracking performance of the controlled piezoelectric and rejects the movement of the coarse stage [25]. The hysteresis and the resonant frequency of the piezoactuator are taken into account by the internal dedicated electronics.

**Observer / Coordinator**  The DISO structure is a dual-stage structure. Therefore the combination of two positions resolves in one single position output, which is to be controlled. This structure is overactuated. An estimate of either one of the positions has to be calculated in order to provide feedback to both loops. The observer provides according to an inverse static gain of the identified piezoelectric actuator model the position of the piezoelectric actuator.

According to the measurement and the estimate of the piezoelectrics position the coarse stages position is evaluated. This estimated coarse stage position provides the feedback to the coarse stage and it provides the deviation of the coarse stage to the reference in feed-forward terms to the fine stage loop [26].

Therefore the coarse stage tracks the filtered reference signal as the integrator in the fine stage controller handles the bias according to nonlinearities and disturbances to the system and the proportional term of the fine stage’s controller fulfills the tracking of the error. In addition the feedforward term of the fine stage loop rejects the coarse stage’s impact to the feedback loop of the fine stage.

**2.3 Controlling fundamentals**

**2.3.1 Kalman estimator**

To design a state feedback controller one has to know the state variables of the system. As shown in 2.2 the dual stage design contains just one measurement, the combined position output signal. When the system input is known and the system is observable or the unobservable states are stable, then the state variables can be estimated by an observer or estimator from this output signal.

The basic scheme of an observer, the Luenberger observer, can be found e.g. in [24]. For the nanometer accuracy and action range, the system is very susceptible to perturbations. Hence it is proposed to implement an observer which is able to reduce disturbances on the system, the Kalman filter.
Kalman filter  The Kalman filter is a convergent observer for the system states of a process described by linear stochastic differential equations. The basic structure of the estimator (2.1) is similar to an ordinary observer with the difference that the observer gain $K_{KF}$ of the Kalman filter is optimal in terms of minimizing a square error [2], as developed in this section. This estimator structure is illustrated in figure 2.5.

$$\dot{x} = Ax + Bu + K_{KF}(y - C\hat{x})$$  \hspace{1cm} (2.1)

wherein $x$: state vector

$\dot{x}$: estimated state vector

$u$: input vector

$y$: measured output position

![Figure 2.5: Observer structure corresponding to (2.1)](image)

The discrete formulation of the filter explains very well the principles of its origin. Therefore the filter equations and working principles are explained in discrete time and in the end of this section the continuous application of the filter, the Kalman-Bucy Filter, is given. If not marked differently, the following remarks are taken from [15] and [35].

As stated above, the filter acts on a stochastic process, which is perturbed by zero-mean white gaussian noise that can be disturbing the internal characteristics or the measurement output of the system. To represent that noise we assume that the
process of the system (2.2) is perturbed by the process noise $w$ and the measured output (2.3) is disturbed by the measurement noise $v$.

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (2.2)$$
$$z_k = Cx_k + v_k \quad (2.3)$$

The process noise has the covariance $W$ and the measurement noise the covariance $V$.

The Kalman filter estimates the states of this process in two steps, the “Time Update”-step and the “Measurement Update”-step. In these two steps the filter calculates a gain $K_k$ that minimizes the a posteriori estimate error covariance $P_k$, with the estimate error $x_k - \hat{x}_k$ as expected value.

$$P_k = E\{[x_k - \hat{x}_k][x_k - \hat{x}_k]^T\} \quad (2.6)$$
$$p(x_k | z_k) \sim N(\hat{x}_k, P_k) \quad (2.7)$$

A minimization of $P_k$ leads to a smaller estimation error and therefore more reliable results. As the variance is updated each step it is dynamic. Thus the filter gain $K_k$ is dynamic as well.

The dynamic filter gain $K_k$ converges to a steady solution. This solution can be chosen as the time-invariant static gain $K$. But when it is used, large deviations of the estimate error in the first samples will occur.

**Prediction** During the “Time Update”-step the filter predicts the process states and the error covariance a priori, as given in (2.8) and (2.9). The development of these equations ((2.8) - (2.12)) as well as the following equations of this section are shown in [15].

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \quad (2.8)$$
$$P_k^- = AP_{k-1}A^T + W \quad (2.9)$$

**Update** The prediction of the state ahead is followed by a correction according to the noisy past measurements. This “Measurement Update” provides feedback to the a priori estimates to obtain an improved a posteriori estimate.

$$K_k = P_k^-H^T(HP_k^-H^T + V)^{-1} \quad (2.10)$$
$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (2.11)$$
$$P_k = (I - K_kH)P_k^- \quad (2.12)$$

When the cycle completed these two steps the previous a posteriori estimates are taken into account in the following a priori estimate of the next cycle.

The user has basically two design features to interact with the filter’s behavior. These are the choice of the process noise covariance $W$ and the sensors noise variance.
V. Usually the V matrix is selected according to the sensors data sheet. W is more difficult to be chosen. Usually its values are based on guesswork and trial-and-error procedures as most of the time the process noise is unknown.

When W is small compared to V that means that the models estimated states are reliable and the measurements are very noisy in comparison. On the other hand as W is large, compared to V, the filter takes less previous samples into account, as the measurements are more reliable than the estimates and previous estimates.

**Extended Kalman Filter** If the System is nonlinear, the Filter has to be varied to be applicable. The solution of the Kalman gain then is not going to be optimal for the nonlinear system but for a linearized estimation.

The modification of the filter to suit nonlinear stochastic differential equations as,

\[
x_k = f(x_{k-1}, u_{k-1}, w_{k-1})
\]

(2.13)

\[
z_k = h(x_k, u_k),
\]

(2.14)

is called Extended Kalman Filter.

In order to insert the nonlinear equations in the filter the equations are to be linearized. The linearization is chosen according to the following Jacobians F, M, H and N of the nonlinear differential equations.

\[
F_{[i,j]} = \frac{\partial f_i}{\partial x_j} \bigg|_{\hat{x}_k,u_k,w=0}
\]

(2.15)

\[
M_{[i,j]} = \frac{\partial f_i}{\partial w_j} \bigg|_{\hat{x}_k,u_k,w=0}
\]

(2.16)

\[
H_{[i,j]} = \frac{\partial h_i}{\partial x_j} \bigg|_{\hat{x}_k,u_k,v=0}
\]

(2.17)

\[
N_{[i,j]} = \frac{\partial h_i}{\partial v_j} \bigg|_{\hat{x}_k,u_k,v=0}
\]

(2.18)

The equations of the Extended Kalman Filter are stated in figure 2.6. This figure shows the cycle of prediction and update as well as the use of the a posteriori updated parameters of the earlier step in the current step of the a priori estimate. Further remarks on the Extended Kalman Filter are to be found in 15 and 35.

**Kalman-Bucy Filter** Similar to the discrete representation the filter can be applied to a continuous-time random process, described by,

\[
\dot{x}(t) = F(t)x(t) + G(t)w(t)
\]

(2.19)

\[
z(t) = H(t)x(t) + v(t)
\]

(2.20)

\[
Ew(t) = Ev(t) = 0.
\]

(2.21)

The continuous optimal Kalman gain is then defined by

\[
K(t) = \lim_{\Delta t \to 0} \left[ \frac{K_{k-1}}{\Delta t} \right] = P(t)H^T V^{-1}.
\]

(2.22)
2 BACKGROUND

Initial estimates for \( \hat{x}_{k-1} \) and \( P_{k-1} \)

**TIME UPDATE ("Predict")**

1. Project the state ahead
   \[ \hat{x}_{k} = f(\hat{x}_{k-1}, u_{k-1}, 0) \]
2. Project the error covariance ahead
   \[ P_k = F_k P_{k-1} F_k^T + M_k W_{k-1} M_k^T \]

**MEASUREMENT UPDATE ("Correct")**

1. Compute the Kalman gain
   \[ K_k = P_k H_k^T (H_k P_k H_k^T + N_k V_{k-1} N_k^T)^{-1} \]
2. Update estimate with measurement \( z_k \)
   \[ \hat{x}_k = \hat{x}_{k} + K_k (z_k - h(\hat{x}_{k}, 0)) \]
3. Update the error covariance
   \[ P_k = (I - K_k H_k) P_k \]

Figure 2.6: Extended Kalman Filter cycle of Prediction and Update

Whereas the error covariance \( P \) in the continuous time domain is obtained by solving the matrix Riccati differential equation (2.23).

Methods to solve Riccati differential equations are explained in [15] and [24].

\[
\dot{P}(t) = F(t)P(t) + P(t)F^T(t) + G(t)W(t)G^T(t) - \bar{K}(t)V(t)\bar{K}^T(t) \tag{2.23}
\]

With the update stated in (2.24), equations (2.22) and (2.23) define the continuous-time Kalman estimator, the Kalman-Bucy filter.

\[
\dot{x} = F(t)\hat{x}(t) + \bar{K}(t)[z(t) - H(t)\hat{x}(t)] \tag{2.24}
\]

The well-known equations of the Kalman filter have been altered to suit many different purposes, some modifications can be found e.g. in [15].

Even if the Kalman filter might provide a very good estimate to the real process, it still relies on a model. Thus the estimate will just be as “good” as this model represents the reality.

### 2.3.2 Optimal control

Optimal control is based on the optimization of a cost functional. The motivation for this approach arises from the increasing complexity and abstraction that comes with multivariable models. Not all variables can be reasonably placed by classical control attempts, they might have no physical equivalent or single characteristics deliver no satisfying results for the whole multivariable plant. Therefore the assessment criteria is placed in a cost functional which is used to design the controller gain and influences the whole trajectory of the system rather than single values. The following development of the optimal controller gain follows, if not differently marked, [24].
The cost functional Because of mathematical terms the easiest cost functional is a quadratic one. The most important characteristics of the system are based on the controlled signal and the controlled output.

Therefore we are now dealing with a linear quadratic cost functional on \( u \) and \( y \), the linear quadratic regulator problem.

Weighting the actuating energy in the cost with \( R \), the deviation during the run with \( Q_y \) and the offset at the final time \( t_e \) with \( S \), leads to the quadratic cost functional \( J \), as in

\[
J = y(t_e)^T S y(t_e) + \int_0^{t_e} y(t)^T Q_y y(t) + u(t)^T R u(t) dt. \tag{2.25}
\]

One could now calculate all possible combinations \( i \) of the input \( u_i(t) \) and compare the size of the cost functionals \( J_i \) in order to find the minimum. But this is not practicable and further the dynamic problem of the closed loop optimization is considered rather than the open-loop determination of the input vector.

Minimizing \( J \) according to the loop closed by \( u(t) = -K x(t) \), rather than minimizing \( J \) in terms of the feed-forward control \( u^*(t) \), leads to the optimization of \( K \). Further the optimum should be time invariant because the control loop should be running continuously in contrast to a fixed-period time interval as stated in (2.25).

This yields the time variant cost functional, as

\[
J = \int_0^\infty y(t)^T Q_y y(t) + u(t)^T R u(t) dt, \tag{2.26}
\]

where \( \min_K J \) depending on \( x_0 \) is to be found.

Linear quadratic regulator The linear, time-invariant control law is independent of \( x_0 \) and can be found by solving the static optimization problem \( \min_K J \) for a linear plant.

When the position \( y \) is substituted by the state vector \( x \), the cost functional is represented by,

\[
J = \int_0^\infty x(t)^T Q x(t) + u(t)^T R u(t) dt. \tag{2.27}
\]

Considering the state equation of the closed loop plant as in (2.28), a different representation of the cost functional is obtained by (2.29)

\[
\dot{x}(t) = \bar{A} x(t)
\]

with \( \bar{A} = (A - BK) \)
$J = x_0^T P x_0$ \hspace{1cm} (2.29)

whereas $P = \int_0^\infty e^{A^T t} \dot{Q} e^{A t} dt$

and $\dot{Q} = Q + K^T R K$

The solution of $P$ can be found by solving the Ljapunov equation (2.30).

$A^T P + P A = -\dot{Q}$ \hspace{1cm} (2.30)

This equation has one symmetric, positive definite solution for $P$, if $Q$ is symmetric positive definite and $A$ is asymptotically stable.

The necessary optimality condition is that the partial derivative of $J$ with respect to all elements $k_{ij}$ of the controller matrix $K$ is equivalent to 0 [24]. Then an optimal controller $K^*$ is given in (2.31).

$K^* = R^{-1} B^T P$ \hspace{1cm} (2.31)

Inserting (2.31) in (2.30), leads to the matrix Riccati equation (2.32).

$A^T P + P A - P B R^{-1} B^T P + Q = 0$ \hspace{1cm} (2.32)

**Linear quadratic regulator equations** The linear quadratic regulator (LQR) for a controllable plant in the continuous time domain is hence defined by:

$u(t) = -K^* x(t)$

$K^* = R^{-1} B^T P$

$A^T P + P A - P B R^{-1} B^T P + Q = 0$.

Because of the optimal controller’s nature, to reduce all state variables to zero, the controller always yields a stable loop and automatically provides, if the previous stated conditions are fulfilled, good robustness properties and phase margins larger than 60 degrees [24].

### 2.3.3 LQG

As shown in the previous sections 2.3.1 and 2.3.2, optimal controller and Kalman filter both rely on the minimization of a cost functional and the solution is given by solving a matrix-Riccati-equation.

When combining these two components on a linear system, the linear quadratic gaussian (LQG) regulator is designed as shown in figure 2.7.

Because of their properties and Riccati origin each feedback loop, the Kalman filter’s and the optimal regulator’s, is robust and stable. According to the separation principle (2.33) the combination of both loops leads to the union of the closed-loop dynamics [31].

$$\frac{d}{dt} \begin{bmatrix} x \\ x - \hat{x} \end{bmatrix} = \begin{bmatrix} A - BK^* & BK^* \\ 0 & A - KC \end{bmatrix} \frac{d}{dt} \begin{bmatrix} x \\ x - \hat{x} \end{bmatrix} + \begin{bmatrix} v \\ w \end{bmatrix} \hspace{1cm} (2.33)$$
Why an LQG control approach? The main issue of designing an LQR controller is to achieve guaranteed stability and to consider the full trajectory of the controlled variables. The Kalman filter is used to estimate the state variables under the presence of zero-mean white gaussian noise.

The main issues for applying the LQG rather than a regular state controller with a Luenberger observer was noise and disturbance reduction by the Kalman filter. The optimal state feedback instead of output feedback was applied to achieve higher performance when acting on every state and not just the system output.

A further objective is to consider a different approach for actuator saturation when using the optimal controller. In addition the consideration of the whole trajectory should give better control of the power consumption in order to reach the desired requirement. Furthermore, the LQR design should be more robust than two decoupled SISO designs, and hence less likely to be influenced by parameter variations due to modeling uncertainties.

Robustness of the LQG Even if every feedback loop has very good robustness properties, the combination of them does not offer any guarantee for stability margins. As [13] stated it: “there are none”. This can be understood when looking at figure 2.7. The robustness of the Kalman filter is available at marker 4 in figure 2.7 and the robustness of the LQR at marker 3. But the transfer functions from the plants output to input (markers 2 and 1 in figure 2.7) is complex. The presence of the Kalman filter couples the two loops and the points of stability and robustness are in the LQG at points where they are not needed for the controller design [31].
2.3.4 Dual-stage tracking control

The dual-stage concept as found in the DDL has been applied in the control of Hard-disk drives (HDD). The disk drive read/write head is positioned over a track within µm accuracy. The two stages here are the arm, onto which the head is mounted and which is driven by a voice-coil motor for the coarse movement, and a microactuator that is used to precisely position the head over the reference track.

According to the increasing growth of the HDD market in the end of the 1990s and the rising accuracy requirements of data density many approaches have been taken to control this special DISO structure.

These approaches include SISO- and MIMO - control structures. A very important point is the distribution of the tracking signal on the two stages. [1]

Control approaches The SISO control approaches either regard the dual stage system as a sequence of SISO systems and design a controller by frequency shaping techniques, or decouple the two stages and regard them as two independent systems. The problems with these techniques are basically the unassured stability and the failure to reach the specified target performance [1].

Modern multivariable control approaches have been applied in order to increase stability margins, gain robustness towards parameter variations, enhance the tracking performance and raise the systems bandwidth.

Most optimal control approaches are based on µ-synthesis and H∞-control which include uncertainties on modeling, and lead to increased track follow, very fast track seek and improved settling performance [21].

Further, the previous discussed LQG technique was applied to the dual-stage design. Because of missing robustness guarantee and stability margins, Loop Transfer Recovery (LTR) was used, as it can be found in [17] and [34]. The desired frequency characteristics of the tracking performance is formulated according to the target feedback loop and is achieved by choosing the process noise and measurement noise covariance matrices of the Kalman filter. The LTR procedure recovers this design at the output of the plant by designing a linear quadratic regulator. The big advantage of this method is the avoidance of trial-and-error pole placement procedures [34].

The disadvantage of the optimal controllers developed in the previous mentioned works is the high order of the controllers, and hence the difficult implementation in practice [1], [21].

Tracking In HDD application the coarse stage actuator is the main tracking actuator. Therefore the voice-coil actuator usually tracks the reference while the microactuator is tracking an estimate of the relative position error between coarse and fine stage, e.g. [20], [32].

When multivariable state feedback controllers are applied a term is needed to split the one dimensional reference signal between the two stages. In [32] a command matrix is proposed to split the signal, but the structure of this matrix is not mentioned.

Further, when the plant is regarded as a multivariable plant, rather than two single stages, output feedback is applied.

Basically, two possibilities of feeding the tracking signal to the loop have been applied. The most common approach is to subtract the feedback signal from the tracking
signal and feed the resulting state error to the controller. Another possibility is to feedforward the reference to the controller’s voltage output in terms of a feedforward gain. [21] states that the feedforward term on the voltage output of the piezoelectric actuator increases the error tracking performance in respect to satisfactory time and frequency responses.
Chapter 3

Modeling the system

The controller which is to be developed in chapter 4 requires a mathematical model of the real system.

This chapter is going to develop this model step-by-step with respect to the dual-stage structure. The model of the coarse stage is developed in section 3.1, followed by the model of the fine stage in section 3.2. It finishes by merging the single models into one overall system model for the DDL in section 3.3 and analyzes its properties in section 3.4.

3.1 Coarse stage

The coarse stage consists of the blade guiding system, the lead screw and the stepper motor. The motor acts on the lead screw, which transforms the rotational motor movement into a linear movement. The sledge of the lead screw is fixed to the blade guiding system on which the cat’s eye including the optics is mounted.

When modeling the system as a whole, the blade guiding system can be taken into account as additional inertia on the steppers equations of motion. This influence is included in the model parameters of the stepping motor when identifying the overall system according to these equations. In addition to that, the blades are bend in order to guarantee horizontal movement, hence the stiffness of the blades results in a force towards the direction of movement. Furthermore play and static friction of the lead-screw influence the coarse stages motion. Modeling these effects is out of the scope of this thesis. If the static friction needs to be modeled or if further restrictions have to be considered will turn out, when the controller is implemented on the real system. If necessary the modifications will be discussed in chapter 6.

This discussion shows that it is reasonable to model the coarse stage according to the motor’s equations with modified parameters. Further, a constant factor which translates the rotational output into a linear motion is added.

3.1.1 Permanent Magnet Stepper Motor

Permanent magnet (PM) steppers have a cylindrical rotor which is radially equipped with pole-alternating magnets. The stator contains windings which each are connected to a set of poles and can be activated by powering the associated phase. If not supplied with electricity the poles close the magnetic circuit and a holding torque develops.
The stepper motor used in the existing prototype is a PM variable reluctance (VR) hybrid stepping motor as shown in 3.1 and found in [33]. It combines the properties of PM and VR stepping motors and is commonly used because of its high mechanical power alongside small step angles [29]. Basically the PM motor is a synchronous motor with stepwise phase switching. Therefore it can also be operated by applying alternating voltage on the phases, which cause the motor to turn.

![PM-VR Hybrid Stepping Motor](image)

Figure 3.1: PM-VR Hybrid Stepping Motor [10]

The common model of a Permanent magnet stepper motor contains four equations, two electrical equations (equations (3.2) and (3.3)) and two mechanical equations (equations (3.1) and (3.4)) [5], [9], [18], [25].

\[
\frac{d\omega}{dt} = -K_m i_a \sin(N_r \theta) + K_m i_b \cos(N_r \theta) - F_f \omega + T_L(\theta) \tag{3.1}
\]

\[
L_i \frac{di_a}{dt} = -R_i a + K_m \omega \sin(N_r \theta) + u_a \tag{3.2}
\]

\[
L_i \frac{di_b}{dt} = -R_i b + K_m \omega \cos(N_r \theta) + u_b \tag{3.3}
\]

\[
\frac{d\theta}{dt} = \omega \tag{3.4}
\]

In these equations \( K_m \) is the motor torque constant, \( F_f \) the viscous friction, \( J \) the inertia of the system and \( T_L \) is the torque load varying over the current position. Furthermore is \( L \) the self-inductance of each phase, \( R \) is the stator-circuit resistance, \( i_a \) the current in winding \( a \) and \( u_a \) is the voltage applied on \( a \). Analogously is \( i_b \) the current in winding \( b \) and \( u_b \) the voltage applied to \( b \). The rotor consists of \( N_r \) teeth.

The calculation of the coarse stages translational output \( y_c \) from the angular motor output \( \theta \) is determined by \( \gamma \) as in (3.5).

\[
\gamma = \frac{10.571[\mu m]}{1.8[deg]} \cdot \frac{360[deg]}{2\pi[rad]} \approx 336.4854 \frac{[\mu m]}{[rad]} \tag{3.5}
\]

### 3.1.2 Parameter identification

The equations (3.1) - (3.3) contain three completely unknown (\( K_m, F_f, J \)) and two uncertain parameters (\( R, L \)). For the latter datasheet values are known but to increase
the degrees of freedom in order to improve the estimation, these parameters are included in the following parameter identification.

The identification signal on the system is a simple voltage step on the \( b \)-phase of the motor whereas the \( a \)-phase is not excited. The measured response of the system is displayed as the dotted line in figure 3.2.

A cost functional \( C \) is chosen to rate the qualification of the tested parameter configuration. It is based on the output value of the measurement and the translational output of the tested model. A squared functional is selected to "punish" large deviations more than small deviations. Therefore the cost functional is given by

\[
C = \int (y_{\text{meas}}(t) - y_{\text{mod}}(K_m, F_f, J, R, L, t))^2 \, dt.
\]

Further as the data is a sampled measurement, the functional has to compare the measured sample with a discrete data set obtained by evaluating the model at fixed time steps. Hence (3.6) describes the used cost functional.

\[
C = \sum_{j=0}^{l} (y_{\text{meas}}(k) - y_{\text{mod}}(K_m, F_f, J, R, L, k))^2 \tag{3.6}
\]

The five parameters are very sensitive to modifications. The inertia \( J \) and the friction \( F_f \) highly deviate from common stepper motor parameters because of the influence of the blade guiding system. An initial guess for these parameters is very uncertain. Applying a standard MATLAB\textsuperscript{®}-algorithm \texttt{fmincon} resolves quickly into a very unsatisfying result, that approaches the slope and the first overshoot sufficiently, and afterwards stays on the mean of the oscillating response. This configuration seems to be a local minima of the cost functional. Varying the initial parameters manually is unlikely to result in an improvement, because the parameters are too sensitive.

As a result of this an evolutionary algorithm, which is not as prone to get stuck in local minima, as the standard algorithm, has been chosen. The MATLAB\textsuperscript{®}-implementation of this algorithm, \texttt{evolution}, has been developed in [22].

The selected evolutionary principle allocates a predefined number of parameter sets (parameter vectors) in a predefined range of the parameter’s values according to a gaussian distribution, each so created vector is called a parent. In the next step the algorithm mutates the parameter values of each parent in a certain range and normally distributes a defined number of children, the offspring, in this area. Afterwards all cost functionals of the offspring are evaluated and ranked. The „fittest“ members, according to the value of the cost functional, are chosen to be parents of the new generation developed in the next iteration. The algorithm proceeds until the derivative of the cost function reaches a certain value, or until a certain number of iterations has been calculated.

By varying e.g. the range, the size of the parent- and offspring-generation, as well as the variance of the gaussian normal distribution in the parameters range the designer has many opportunities to influence the evolution of the algorithm.

Further, the above-mentioned procedure is just one special case of an evolutionary algorithm. Different selection and recombination principles are available. The selected algorithm is a mutational genetic algorithm that involves no cross-overs in the parents-generation and includes no pre-selection to the size of the offspring generated by a
parent, hence every parent creates the same size of offspring [22].

When selecting a large variance, large parameter ranges and a high parent- and offspring population, the algorithm evaluates many possible solutions. According to the large distribution of individuals over the complete five-dimensional space the algorithm leaps over many local minima, as there is most likely another individual on a “fitter” area of the cost functional, which is closer to the global minima.

![Identification signal](image)

Figure 3.2: Identification of stepping motor parameters

With this algorithm a set of parameters has been evaluated. Important for the obtained model is the dynamic rather than the continuing oscillations to the end of the identification signal. Therefore the samples inserted in the cost functional have been reduced to not cover the full 1 second response (8000 samples) but ~0.2 seconds which are characteristic for the dynamic step behaviour.

The evaluated parameters take the dynamics of the system, especially the slope and the first oscillations very well into account. Additionally, the frequency of the oscillations matches good, as illustrated in the very small phase difference in figure 3.2. The damping of the oscillation is still larger than in the measured data, but this can be due to energy introduced by the blade guiding system, and will presumably not occur in synchronous drive as the motor is not operated in steps.

The parameters obtained by the algorithm are stated in table 3.1. The result shows the parallel connection of two resistors of 1 Ω each (datasheet: 1 Ω) and the parallel connection of two inductions of 1.2 mH each (datasheet: 1.5 mH).
3 MODELING THE SYSTEM

\[ R = 0.5 \ \Omega \]
\[ L = 0.0024 \ \text{H} \]
\[ K_m = 0.5678 \ \text{N m/A} \]
\[ F_f = 0.0017 \ \text{N m s/rad} \]
\[ J = 0.0013 \ \text{kg m}^2 \]

Table 3.1: Identified Parameters

3.1.3 Linearization

The model in hand is nonlinear as evident from the \( \sin(\theta) \) and \( \cos(\theta) \) terms multiplied by \( \omega \) in equations (3.1) to (3.3). To obtain a linear system for the linear controller design the model has to be transformed.

The simplest approach to a linear model is the linearization by building the jacobians and solving them at each operation point. This would lead to a time variant model, which will increase the computing effort of the system and the effort of the controller design.

Another possibility is the small angle approximation which would resolve the time variant problematic. According to the fact that the controller might switch the controlled voltage rapidly, the motors axis and the active phase may vary largely, the motor may fall out of phase. This approach presents strong limitations to the performances and another approach should be seeked.

Thus an alternative linearization technique, the nonlinear DQ-transformation (Park transformation) with nonlinear feedback, is applied [3], [4], [5].

The DQ-transformation is given by (3.7) and (3.8).

\[
\begin{bmatrix}
i_d \\
i_q
\end{bmatrix} = \begin{bmatrix}
\cos(N_r \theta) & -\sin(N_r \theta) \\
\sin(N_r \theta) & \cos(N_r \theta)
\end{bmatrix}
\begin{bmatrix}
i_a \\
i_b
\end{bmatrix}
\]

(3.7)

\[
\begin{bmatrix}
v_d \\
v_q
\end{bmatrix} = \begin{bmatrix}
\cos(N_r \theta) & -\sin(N_r \theta) \\
\sin(N_r \theta) & \cos(N_r \theta)
\end{bmatrix}
\begin{bmatrix}
v_a \\
v_b
\end{bmatrix}
\]

(3.8)

Applying (3.7) and (3.8) to (3.2) and (3.3), and substituting the result in (3.1) resolves in (3.9), (3.10) and (3.11).

\[
\frac{d\omega}{dt} = \frac{K_m}{J} i_q - \frac{F_f}{J} \omega - \frac{T_L}{J}
\]

(3.9)

\[
\frac{d}{dt} i_d = \frac{v_d}{L} - \frac{R_l i_d}{L} + N_r \omega i_q
\]

(3.10)

\[
\frac{d}{dt} i_q = \frac{v_q}{L} - \frac{R_l i_q}{L} - \frac{K_m}{L} - N_r \omega i_d
\]

(3.11)

The nonlinear feedback will linearize this system. An inspection of the equations suggests to use (3.12) and (3.13) as nonlinear feedback, as also suggested in [5].

\[
v_d = R_l i_d - N_r L \omega i_q + Lu_d
\]

(3.12)

\[
v_q = R_l i_q - N_r L \omega i_d + K_m \omega + Lu_q
\]

(3.13)
With these transformations the system becomes

\[ \frac{d\omega}{dt} = \frac{K_m}{J^2} i_q - F_f \frac{J}{J} \omega - \frac{T_L}{J} \]  
(3.14)

\[ \frac{d\theta}{dt} = \omega \]  
(3.15)

\[ \frac{di_d}{dt} = u_d \]  
(3.16)

\[ \frac{di_q}{dt} = u_q \]  
(3.17)

The continuous linear state space representation with \( \theta, \dot{\theta}, i_d, i_q \) as state variables, and \( u_d \) and \( u_q \) as input variables is given by

\[
\begin{bmatrix}
\dot{\theta} \\
\dot{i}_d \\
\dot{i}_q
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & -\frac{F_f}{J} & 0 & K_{ps} \\
0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\theta \\
\dot{\theta} \\
i_d \\
i_q
\end{bmatrix} +
\begin{bmatrix}
0 & 0 \\
0 & 1 \\
1 & 0 \\
0 & 1
\end{bmatrix}
\begin{bmatrix}
u_d \\
u_q
\end{bmatrix}
+\begin{bmatrix}
0 \\
-T_L \\
0 \\
0
\end{bmatrix}
\]  
(3.18)

\[ \begin{bmatrix}
\theta \\
\dot{\theta} \\
i_d \\
i_q
\end{bmatrix} = \begin{bmatrix}
1 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\dot{\theta} \\
i_d \\
i_q
\end{bmatrix} + \begin{bmatrix}
u_d \\
u_q
\end{bmatrix}
\]  
(3.19)

### 3.2 Fine stage

Piezoelectric ceramics generate an electric potential when pressure is applied and inversely expand when an electric potential is supplied. This effect is used in actuators that have high operating forces with high dynamics at a relatively short stroke [29].

The piezoelectric actuator in hand is a tripod design of three parallel piezoelectric stack actuators. These can tilt the platform (up to 5 mrad) and allow a translational stroke of up to 30 \( \mu \)m. The three actuators are controlled with a dedicated servo-controller, consisting of an analog proportional integral controller. It shall compensate for the drift and hysteresis of the actuators.

Therefore it is not reasonable to model the fundamental piezoelectric equations. Moreover the piezoelectric actuator can be seen as a black box and is identified according to standard identification procedures.

For this identification a step signal with varying step duration is applied to the system. The hysteresis of the piezoelectric is still present but in the low expansion range, where the piezoelectric is to be operated in the later tracking problem, the relation between expansion and stress is almost linear. Therefore, the input voltage for the identification is as well selected in this range. This should resolve in a very accurate model with a representable static gain.

The identification process with the MATLAB®-identification-toolbox resolves in an ARX model with \( na = 3, \ ng = 1 \) and \( nk = 2 \). ARX models are described as \( Ay = Bu + e \), where they characterize models with an autoregressive (AR) part \( A(q)y(t) \) and an extra \((X)\ input \ B(q)u(t) \). The terms \( na \) and \( nb \) refer to the order of the polynomials \( A(q) \)
and $B(q)$, as $nk$ is the dead time of the system [6], [23].

The stated ARX model was selected according to the dynamics and most importantly the bandwidth of the system. The bandwidth can be analyzed in a Bode plot and is the range of frequencies at which the signal’s Fourier transform has a power above 50% of the maximum value. In the Bode plot this is the value at which the plots magnitude falls below -3dB of the maximum value. Figure 3.3 shows the identified systems bandwidth at 269Hz in the Bode plot.

![Bode magnitude of identified ARX 312 model](image)

Figure 3.3: Bode plot of the identified model

The selected models bandwidth is lower than the bandwidth obtained in earlier identification processes. This is most likely due to the very robust tuning of the internal controller of the piezoelectric actuator. Still the bandwidth of 269 Hz is higher than the required bandwidth of 250 Hz (cf. section 2.2.1) and will be sufficient for the controller development.

The identified model produces very good results when cross validating the deviation with an additional data set and is therefore selected.

The model obtained is the following discrete time model

$$A(q)y = B(q)u + e$$  \hspace{1cm} (3.20)

wherein, $A(q) = 1 - 2.442q^{-1} + 2.059q^{-2} - 0.5995q^{-3} + 0.5087q^{-4}$,

and $B(q) = 0.04901q^{-2}$.

This model is transformed into the continuous time domain and assembles to the state space representation of the fine stage, as in (3.21) and (3.22). It contains $n = 3$ state variables, $r = 1$ output and $m = 1$ input.

$$\dot{x} = A_f x + B_f u$$  \hspace{1cm} (3.21)

$$y = C_f x + D_f u$$  \hspace{1cm} (3.22)
3.3 Overall model

As discussed in section 2.2 the two systems are to be combined on the same output. This composition structure is illustrated in figure 3.4.

Therefore the state space representation of the system is composed out of the single systems state space representations. The mass difference of the two stages as well as the operating bandwidth differ largely, which means that all couplings between the two systems are very small. Therefore the couplings are neglected which leads to the system matrix composition, where the coupling matrices $A_{cf}$ and $A_{fc}$ are equal to a corresponding number of zeros $0$, and $A$ is an $n \times n$ matrix. Further there are no interactions between the inputs. This leads to the diagonal composition of the single systems input matrices $B_c$ and $B_f$ and an equivalent number of zeros on the remaining values of $B$ $(n \times r)$. The overall models output matrix $C$ of the dimension $r \times n$, is the combination of the outputs of the two systems on the same overall output vector. This is achieved by connecting the matrices in parallel. As no feedthrough terms are available in the system the feedthrough matrix $D$ $(r \times m)$ is $0$.

$$A = \begin{bmatrix} A_c & 0 \\ 0 & A_f \end{bmatrix}$$ (3.23)

$$B = \begin{bmatrix} B_c & 0 \\ 0 & B_f \end{bmatrix}$$ (3.24)

$$C = \begin{bmatrix} C_c & C_f \end{bmatrix}$$ (3.25)

$$D = 0$$ (3.26)

Note that the index $c$ refers to the state space representation of the coarse stages and not the continuous-time domain, as the index $f$ refers to the state space representation of the fine stage. So the overall systems state space representation is

$$\dot{x} = Ax + Bu$$ (3.27)

$$y = Cx + Du.$$ (3.28)
3.4 System properties

In this section, the structural properties of the system are analyzed. These properties include the relation between the models input, output, states and disturbances. First of all, the stability of the system is to be determined.

**Stability** A continuous time system is considered „stable“ if no real part of the eigenvalues of the system matrix $A$ is greater than zero [24].

The eigenvalues of the system are stated in Table 3.2. As there is no unstable mode in the system (all real values are 0 or negative), the system is stable. Even if not asymptotically stable because it contains poles at zero.

$$
\begin{array}{c}
\lambda_1 = -1296.4 \\
\lambda_2 = -1398.4 + 2589.8i \\
\lambda_3 = -1398.4 - 2589.8i \\
\lambda_4 = 0 \\
\lambda_5 = -1.3 \\
\lambda_6 = 0 \\
\lambda_7 = 0
\end{array}
$$

Table 3.2: Eigenvalues of the model

**Controllability** Controllability determines the ability of the system to reach every possible state configuration by an appropriate input signal. This means the states are reachable from the input [2].

The controllability for a multivariable model according to the Kalman’s theorem of controllability (3.29) is given, if the rank $r$ of the controllability matrix $C_{control}$ is equal to the number of system states $n$.

$$
C_{control} = \begin{bmatrix} B & AB & A^2B & \ldots & A^{n-1}B \end{bmatrix}
$$

In this equation $A$ and $B$ are the system matrix and the input matrix. The system in hand contains seven states, the computation of the rank $r$ of $C_{control}$ resolves in 3. As $r \neq n$, the system is according to the Kalman criteria not controllable.

This was not expected. A check on the individual systems shows that each system itself is controllable. The linear composition should not alter this property. Evaluating the condition of the matrix $C_{control}$ offers a large discrepancy between the maximum and minimum of its values. This is very likely the source of the erroneous result. Hence a different theorem is analyzed to verify the stated thoughts.

The controllability criteria of Hautus [24] states that, if (3.30) is valid for all eigenvalues $\lambda_i$ of the system $(A, B)$, the system is controllable.

$$
\text{rank} (\lambda_i I - AB) = n
$$

(3.30) is fulfilled for all eigenvalues of $(A, B)$ and the composed system is therefore controllable.
**Observability** Dual to the input-state relation of controllability, observability concerns the state-output relation. Hence it clarifies if every possible state configuration can be computed according to the output signal.

Analogously to the Kalman criteria of controllability a Kalman criteria of observability exists. Therein the observability matrix $O$, as in (3.31), has to be of the same rank $r$ as the system has states $n$.

$$O = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{pmatrix} \quad (3.31)$$

Again the Kalman criteria is not fulfilled which would mean that the system is not observable. Calculating the Kalman criteria for the individual models reveals the unobservability of the steppers model. Therefore the result of the Kalman criteria of observability is very likely to be trusted. In order to verify this result the Hautus criteria of observability is considered.

This criteria is that every eigenvalue $\lambda_i$ of the system matrix $A$ has to fulfill (3.32) for the system $(A,C)$ to be observable.

$$\text{rank} \left( \lambda_i I - A \right) = n \quad (3.32)$$

As expected (3.32) is not valid for the system in hand and the system is not observable. The impact of this result on the controller design and the further proceedings is discussed in chapter 4.
Chapter 4

Multivariable Controller Design

The focus of this chapter is to illustrate the design of the LQG controller. The equations and the theory have already been stated in chapter 2; therefore, this chapter outlines the discussion of the problems encountered when applying the theory to the DDL LQG design.

4.1 Control scheme

The model, described by (3.27) and (3.28), is supposed to be controlled by an optimal controller, as it was described in section 2.3.2. The controller is to be combined with a state estimator.

The first task is to analyze the composition of the systems and to determine how the reference signal is fed to the system. As there is just one sensor available, the loop is closed over the measured position. The estimator is placed in the feedback and the controller in the forward path of the closed loop system as shown in figure 4.1. At this point, the controller would try to make all states estimated by the Kalman filter converge to zero. This is not the aim of a tracking layout hence the reference position has to be added to the loop. This is most likely to be achieved in terms of a prefiltered addition $x^*$ of the reference states between the estimator and the controller. Another alternative is to add the reference to the systems input voltage in terms of $u^*$. These two possibilities are illustrated in figure 4.1. F and N represent transformations to convert the reference into a matching input or state representation.

![Figure 4.1: Tracking controller](image-url)
As the controller reduces its input values to zero, it becomes obvious that the input to the controller and furthermore the treatment of the tracking signal is a key issue of the closed loop tracking design. [16] suggests that both reference inputs, F and N, are to be used in combination. Hence the state reference $x^*$ combined with the estimated state $\hat{x}$ vector resolves in the tracking error $e$. And the voltage feed forward term $u^*$ is designed to guarantee the required input voltage to achieve the tracking result. This tracking scheme is to be used when the model of the reference signal is available [16]. Other references introduce just a feedforward on the plant input value to guarantee the tracking [8], [7]. As the DDL is overactuated, the distribution of the reference signal $y_{ref}$ into the state reference $x^*$ is nontrivial, as there are multiple solutions possible.

Reference input  The corresponding static input vector distribution to a certain reference value $y_{ref}$ is obtained by using the inverse of the static gain (also called DC-gain) from the input $u^*$ to the closed loop systems output $y$ as in (4.1). It is similar to the regular open loop static gain obtained when solving the state space equations with neglected dynamics $\dot{x} = 0$ for $u$. Further, it considers the system matrix of the closed loop $A - BK$ instead of the regular system matrix $A$. The closed loop system corresponds to the system stated in section 2.3.3.

$$\frac{y}{u^*} = \text{DCgain}_{CL} = -C (A - BK)^{-1} B$$  (4.1)

Calculating the corresponding state vector $x^*$ to a certain reference value $r$ is not straightforward. No solution has been found to transfer a single value in a corresponding state representation. The characteristics of the $C$ matrices of each subsystem distribute the reference to only one position-related state, whereas the other states governing this state are hidden. As it is not necessary to insert both terms $x^*$ and $u^*$ to obtain a good tracking result, no further efforts are taken to find a corresponding transformation $F$. The controller will care for the disturbance rejection and the feedforward term $u^*$ will guarantee the tracking.

Considering the special structure of the dual-stage design in tracking terms a static-gain is not sufficient. The multivariable model, that splits the signal on both stages statically would consider the fine stage just on a very small amplitude and would try to fulfill the tracking with the coarse stage only. So the multivariable static split is able to track the reference, but without any consideration to the special structure of the dual-stage design. Hence all benefits of this structure would be lost. Therefore, each subsystem is considered separately rather than the multivariable model. As a result, each subsystem’s inverse DC gain transforms a dedicated reference signal, determined in the next paragraph, into feedforward voltage terms, which are added to the corresponding voltage in the loop after the controllers output.

Dynamic aspects  After determining how to transform a reference signal to a corresponding input signal, the question is how the dynamics can be included in the compensator with respect to an optimal tracking result? The prefilter $N$ is to be dynamic in order to involve each stage and its advantages for the tracking. This problem is very similar to the problems discussed in section 2.3.4 and $N$ can be seen as an extension of
the compensator designs in the decoupled dual-stage controller designs.

In order to distribute the reference to the single stages one position is to be estimated. As the Kalman filter already provides an estimate of the system, the most obvious idea is to use this estimate to provide an additional position to the compensator. This Kalman estimate and therefore the precision achieved are very model dependend. Another possibility is to calculate the coarse stage position according to the inverse piezoelectric model. This is defined by the inverse open loop static gain \( \frac{u_f}{y_f} = -(C_fA_f^{-1}B_f)^{-1} \). Furthermore, the difference between the measurement and the estimated coarse stage position will resolve in the feedforward reference of the fine actuator. With the estimated fine stage position two positions are available and the system can be split to satisfy the dynamic and tracking requirements.

The fine stage is required to act very fast on the signals changes. The coarse stage is designed to care for the slow coarse movement and further to drive the piezoelectric actuator into its neutral position. It seems very straightforward to track the actual reference value \( r \) with the coarse stage and track the error between the reference position \( r \) and the estimated coarse stage position \( \hat{y}_c \) with the fine stage. The coarse stage position-estimate according to the measurement and the inverse piezoelectric model depends on the models accuracy and the measurement itself. This measurement is perturbed by noise which would be fed back to the compensator, and then be tracked by the regulator. This is not desired and, hence, the Kalman estimate of the coarse stage position is used.

To assure the precise tracking of the real value and to neglect all perturbations affecting the system, an integrator term on the estimate error is introduced. This integration term \( i\hat{e} \) adds to the position reference of the piezoelectric. The integrator has a minimum and maximum saturation, so that large steps, which cannot be followed by the system does not saturate the actuator.

\[
i\hat{e} = \int [y_{\text{meas}} - (\hat{y}_c + \hat{y}_f)] dt \bigg|_{\min}^{\max}
\]

Hence the equations of N are (4.2) and (4.3).

\[
u^*_c = r \cdot (\text{DCgain}_{CLc})^{-1} \tag{4.2}
\]
\[
u^*_f = (r - \hat{y}_c + i\hat{e}) \cdot (\text{DCgain}_{CLf})^{-1} \tag{4.3}
\]

**Final scheme**  These aspects are inserted in the control scheme. Including the dq-transformation out of Section 3.1.3, the scheme is given in figure 4.2, whereas the design of the estimator and the regulator are discussed in the following sections.
4.2 Estimator design

The Kalman filter observes the process and returns an estimation of the plants states. The equations of the continuous time Kalman-Bucy-filter stated in section 2.3.1 are recalled hereafter.

\[
\dot{x}(t) = F(t)x(t) + G(t)w(t) \quad (4.4)
\]
\[
z(t) = H(t)x(t) + v(t) \quad (4.5)
\]
\[
\bar{K}(t) = PH^T V^{-1} \quad (4.6)
\]
\[
\dot{P}(t) = F(t)P(t) + P(t)F^T(t) + G(t)W(t)G^T(t) - \bar{K}(t)V(t)\bar{K}^T(t) \quad (4.7)
\]

Usually, the estimator design deals with the treatment of noise through the tuning of the covariance matrices. The design assumes the presence of zero-mean gaussian white-noise. For simulation this noise is to be inserted in the plants equations.

The measurement noise could be measured from the prototype, but for simulation a variance of $V = 10^{-2}$ is applied to the models output. The variance of the process noise has been assumed with the order of magnitude to $10^{-6}$, resulting in a small parameter change, as for the process noise no data is available.

As described in chapter 3, the model equations are not observable. Hence if the model parameters differ from the identified parameters, even a modified Kalman filter will not be able to converge to the “real” values. This turned out to be the main design issue in this case. An equal behaviour occurs when the initial values of filter and plant vary. In an observable system a compensated Kalman filter could be used to reduce this bias. The Kalman filter obtains feedback from the measurement, which is – speaking in SISO terms – coupled through the output matrix to the first state of the coarse stage and to the first state of the fine stage. The filter tries to compensate the error on those two states. As the effect on the unobservable states is not “seen” from the filter, the impact of the “cost” of convergence of the observable states is unknown. But as long as the unobservable states are stable this should not affect the simulation. Simulations
have shown that a bias evolves on all states. The first states bias converges slightly to a smaller value (but never to 0), but the bias of the second state increases. As the unobservable states equations depend on the second state, their estimate is corrupted. The bias value depends on the order-of-magnitude of the weighting matrices.

Several methods have been evaluated for reducing filters bias (e.g. the Compensated Kalman Filter). As only one combined measurement is available, not even these methods can diminish the bias, as there is no certain feedback information available. Additional sensors and hence an observable system would reduce the bias and improve the Kalman filters estimate.

**Choice of V and W** The designer has two covariance matrices, $V$ and $W$, at his disposal to tune the filter. The following figure 4.3 shows very well the impact of the design matrices on the system. The figure shows from left to right the change of the systems behaviour according to an increasing measurement covariance matrix $V$. The dash-dotted line shows the zero-mean noise impact on the systems response. The reference value is fixed to 50 µm. The increase of the measurement covariance matrix $V$ leads to a loss of “trust” in the measurement values. This is shown by the estimate, the dotted-line, and its behaviour towards the actual position, the solid line. In the first picture, the estimate approaches the disturbed response, and therefore the regulator is able to influence the disturbance and drives the actual position to the reference. Increasing $V$ decreases the reaction of the estimator and finally leads to the complete neglection of the measurement and its disturbance. This is shown in the third image as the estimation stays on the reference, as it “trusts” its internal model more than the measurement. The measurement follows exactly the amplitude of the disturbed reference because the regulator does not “see” it and therefore is not able to influence it.

The final tuning of the Kalman filter has to be fulfilled on the real system as the noises are just assumed values. The simulation in chapter 5) underlies the assumption of low process noise and high measurement noise and was optimized therefore.

![Figure 4.3: Varying the Kalman filters covariance matrix](image)

Figure 4.3: Varying the Kalman filters covariance matrix
4.3 Regulator design

The linear quadratic regulator was developed in section 2.3.2. For the sake of simplicity the equations are stated again, as

\[
\begin{align*}
    u(t) &= -K^* x(t) \\
    K^* &= R^{-1} B^T P \\
    A^T P + PA - PB R^{-1} B^T P + Q &= 0.
\end{align*}
\]

As \( K^* \) is constant, the time invariant equations (4.9) and (4.10) can be solved offline. The gain (4.9) is resolved from the solution to the matrix Riccati differential equation (4.10). The Riccati equations are nonlinear and the designer’s impact on the solution is given by the weighting matrices \( Q \) and \( R \). These matrices have to be positive semi-definite. The most obvious action is to choose each matrix diagonal as

\[
Q = \begin{bmatrix}
q_{11} & q_{22} & q_{33} & q_{44} & q_{55} & q_{66} & q_{77} & q_{88}
\end{bmatrix}
\]

and \( R = \begin{bmatrix}
r_{11} & r_{22} \\
r_{22} & r_{33}
\end{bmatrix} \).

The weighting has a physical correspondence as the eight values of the main diagonal of \( Q \) correspond to the weighting on the eight states. Further, the main diagonal of \( R \) weights the three elements of the input vector.

To find an initial guess for the trial-and-error procedure to determine the elements of the weighting matrices, [16] introduces Bryson’s rule as (4.11) and (4.12).

\[
q_{ii} = \frac{1}{\text{maximum acceptable value of } x_i^2}, \text{ for } i \in \{1, 2, ..., n\} \tag{4.11}
\]

\[
r_{jj} = \frac{1}{\text{maximum acceptable value of } u_j^2}, \text{ for } j \in \{1, 2, ..., m\} \tag{4.12}
\]

This corresponds to a scaling of all appearing variables in the quadratic cost functional to 1 and simplifies the following tuning procedure.

The figures 4.4, 4.5 and 4.6 illustrate the impact of the variable change on the systems performance. Figure 4.4 shows the performance of the weighting related to a coarse stages state in a step response. Further, figure 4.5 shows the performance of the weighting related to the states of the fine stage in terms of the Kolmogorov tracking properties. Both figures show that obviously the increasing performance, in terms of a smaller error or a faster response, is counter-balanced with a higher actuator activity. Figure 4.5 shows in the left quadrants of each subfigure the absolute tracking error, and
Figure 4.4: Tuning $Q$ parameters related to coarse stages states according to a step response. Faster response, slightly rising overshoot and increasing servo-action with increasing $q_{11}$.

in the right two quadrants, the input voltage of the piezoelectric. The left quadrants illustrate the performance and the right quadrants the activity of the piezoelectric actuator. As in each subfigure the top row refers to one value of the tuned parameter and the bottom row to another larger value of the parameter. Comparing figure 4.5(a) and figure 4.5(b) shows that the impacts for the same performance improvement and their cost can differ. Therefore it is important to analyze each states influence to the tuning.

The results are more and less obvious, as the physical meanings of the fine stages states are not known, their impact has to be simulated for being judged. The physical meaning of states 1 and 2 are well known, and therefore the increase of weighting on the first state (proportional to the coarse stage position) will lead to a more aggressive behaviour in order to reach the reference value. The increase of weighting on the second variable (the speed of the coarse stage) will slow the system down in order to avoid the large cost in the functional.

Figure 4.6 shows the impact of the weighting of $R$ on the system. As expected a higher activity leads to a higher cost, which is avoided when the corresponding weighting increases. The ratio of the dimensions of $Q$ to $R$ is very important to the systems behaviour. Many books (e.g. [24]) introduce a factor $\rho$, as in $R = \rho \cdot R_{LQ}$, in order to show the impact of the ratio on the performance. This impact is covered in the large variation of dimension in the related figures. Thus, it is obvious that an increase of $\rho$ raises the cost of the input signals in the functional and therefore decreases the performance values.
### Figure 4.5: Tuning Q parameters with impact on fine stage according to Kolmogorov tracking

**Saturation of the actuator** When using the above described impacts on the systems behaviour a controller can be designed to achieve special requirements, e.g. slope time and settling time. In addition, it is possible to design a controller which does not cross the maximum physical values of the actuators – the input voltage – for a certain step response. But as the controllers gain is a static gain the performance and action
range will only be optimized for this reference value. A lower reference value will proportionally lead to a lower controller output voltage and a higher reference value will lead to a higher controller output voltage. To avoid the physical saturation of the actuators a virtual saturation term has been considered in the design on all three voltage inputs to the plant.

4.4 Discretization

The developed LQG controller is designed to be used with the dedicated electronics provided by the ESO. Thus the whole design has to be merged on the microprocessor which operates the data at discrete time steps. Hence, the controller has to proceed a digital input signal, which is to be sampled from the continuous time output signal of the plant in an analog-to-digital (A/D) converter. The controllers digital output is to be transformed in a continuous time signal by a digital-to-analog (D/A) converter.

The A/D converter samples the continuous signal at \( t = k \cdot T \), whereas \( T \) is the sample time and \( k \) the current timestep at which the signal is evaluated. The task of the D/A converter, to create a continuous time signal out of a digital input, is achieved by a zero-order-hold term, which holds the last value over \( T \) until the next value is evaluated. This transformation and the resulting control scheme is illustrated in [4.7].

The linear discretization is usually given by a discrete model, that evaluates the value of the next timestep as in

\[
\begin{align*}
x(k + 1) &= A_d x(k) + B_d u(k) \\
y(k) &= C_d x(k) + D_d u(k).
\end{align*}
\]
A linear model is usually discretized from its continuous time differential equations - that is if $A$ is nonsingular - according to \[ \text{(4.15)} \] as in

$$A_d = e^{AT} = \mathcal{L}^{-1}\{(sI - A)^{-1}\}T = T,$$

$$B_d = \left( \int_{\tau=0}^{T} e^{A\tau} d\tau \right)B = A^{-1}(A_d - I)B,$$

$$C_d = C \text{ and } D_d = D. \quad \text{(4.17)} \quad \text{(4.18)}$$

The control scheme comprises three blocks, that have to be adapted to the discrete scheme: the prefilter, the Kalman estimator and the controller itself.

**Discrete linear quadratic regulator** The controller requires a discretization of the linear model. Thus the steppers DQ-transformed model (3.18) and (3.19) are discretized according to 4.15 and 4.18. The model of the piezoelectric has already been identified by a discrete data set, and therefore a state space representation of the discrete identified model (3.20) is used. A discretization of the complete model in MATLAB® would, due to the condition of the system matrix, lead to an erroneous representation. Combining these two discrete stages provides a model, which can be used to determine the discrete controllers optimal gain, as in (4.19), with the solution of the discrete time Riccati equation (4.20).

$$K = (R + B_d^T PB_d)^{-1}B_d^T PA_d \quad \text{(4.19)}$$

$$P = Q + A_d^T PA_d - A_d^T PB_d(R + B_d^T PB_d)^{-1}B_d^T PA_d \quad \text{(4.20)}$$

**Discrete Kalman estimator** In the Kalman filter the a priori prediction as well as the a posteriori update cycle have to be discretized. The discrete equations have already been developed in section 2.3.1

For the discrete a priori estimate a non-DQ-transformed discrete model of the coarse stage has to be obtained. As in the continuous time, the nonlinear equations (3.1),(3.2) and (3.3) have been used, a discretization of these equations is the most obvious approach. A Runge-Kutta 4th order method (RK4) achieves more precise simulation
results than a standard Euler-forward can provide and is therefore selected \([30]\). For a nonlinear differential equation \(\dot{y} = f(y, u)\) the Runge Kutta method is given by (4.21).

\[
y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4), \text{ wherein (4.21)}
\]

\[
k_1 = f(t_n, y_n),
k_2 = f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2} k_1\right),
k_3 = f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2} k_2\right),
k_4 = f(t_n + h, y_n + hk_3)
\]

The a posteriori correction calculates the update according to the equations in figure 2.6. The jacobians \(M\) and \(N\) are equal to the identity matrix. Further the jacobian \(F\) is again created separately for both stages. For the coarse stage it is defined as (2.15) and builds the partial derivative of the nonlinear stepper equations, that have already been used for the a priori estimate. This derivative is evaluated at each step \(k\) and composed with the already discretized system matrix of the fine stage. \(H\) is created analogously and the equations can be evaluated in the discrete time domain.

**Discrete prefilter** The terms relating to the model in the prefilter equations in the prefilter \(N\) are substituted with their discrete representation. The static gain calculation in the discrete time domain is different from the continuous time one, and is calculated as

\[
\frac{y_k}{u_k} = d\text{DCgain}_{CL} = C(I - (A - BK))^{-1}B. \tag{4.22}
\]

**4.5 Conclusion and further remarks on the controller design**

In this chapter the design of the controller and the related components was discussed. The system output is the combination of the two outputs of the single stages and model of the coarse stage is nonlinear and unobservable. Thus the design of the Kalman estimator turned out to be more complex and due to the linearization lacks precision. As the a priori estimation in the continuous time domain used the continuous nonlinear equations the solution in the discrete domain is always just an approximate of these equations. Furthermore, the design of the controller structure is, according to the dynamic reference feedforward, always oriented on a composition which neglects all couplings between the two stages. The resulting controller is more a combination of two SISO linear quadratic regulators similar to the DISO systems analyzed in section 2.3.4. The simulation in Chapter 5 and implementation in Chapter 6 will show if the design effort taken could improve the performance or not and if the electronics are able to handle the complexity of the multivariable controller.

**Further Remarks** A first idea of this thesis was to develop a new method for the LQR applied to the DISO design to influence certain system properties, as e.g. saturation. This idea was not to be realized due to the high complexity of the solution to
the matrix Riccati equation, as shown hereafter. Considering the saturation with an optimal proportional regulator seems at first view not to be possible. It can always be tuned for a certain range of inputs, but as the gain is proportional it will never be able to act on the full scale over the whole input range. A special structure of the proportional controller gain $K$ might be able to solve this problematic for a dual stage system. The controller gain developed in the previous sections for the dual-stage, is more a linear combination of an optimal gain for the coarse stage and the fine stage. A closer look at its structure

$$K = \begin{bmatrix}
k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} & k_{17} & k_{18} \\
k_{21} & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} & k_{27} & k_{28} \\
k_{31} & k_{32} & k_{33} & k_{34} & k_{35} & k_{36} & k_{37} & k_{38}
\end{bmatrix}, \quad (4.23)$$

shows that the upper left $2 \times 4$ array refers the states of the coarse stage (states 1-4) to the input voltage of the coarse stage ($u_1$ and $u_2$) and the lower right $1 \times 4$ array refers states of the fine stage (states 5-8) to the input voltage of the fine stage ($u_3$). The other components are zero or very small.

If now the controller gain could be used to act on the input vector of the coarse stage according to a state value of the fine stage, saturation of the fine stage actuator could be taken into account. For example, if the state of the piezoelectric differs largely from 0 this means that it is close to its saturation or already in it. Thus, a gain in value $k_{25}$ could then accelerate the coarse stage (as the input voltage $u_2$ refines to the q-voltage, and the state $x_5$ to the position of the piezoelectric) in order to move faster to the desired position and therefore relax the fine stage or even move it to 0. Achieving this special gain structure might be possible by a change of the optimal calculation in the Riccati equations or taken the coupling elements in the $Q$ matrix into account.

The main problem implementing this thought, is the highly complex solution to the Riccati differential equation and therefore, the specific variation of the constant gain.
Chapter 5

Simulation and comparison to the SISO design

5.1 Closed loop properties

To analyze the closed loop properties of the system is not straightforward. Two different linearization techniques have been applied. The Kalman filter is acting on a Jacobian linearization around the current working point of the model and the regulator is considering a dq-transformed state space representation. Analyzing the poles of the closed loop will resolve in poles valid for the dq-transformed model. But the plant is operated in the ab-domain, and the nonlinear retransformation changes these values. Therefore no statement can be made about the analysis of the closed loop properties of the system.

5.2 Simulation of discrete controller

The controller has been tuned according to the systems step response and the Kolmogorov stochastic signal tracking. As described in section 4.3 the regulator parameters responsible for the performance of the coarse stage are easiest to be tuned when the system is performing a step. The parameters corresponding to the fine stages states are best to be tuned performing a Kolmogorov tracking. The Kolmogorov tracking is the benchmark for the later application of the developed control scheme. Thus, a fine-tuning according to this tracking signal determined the final value of the overall parameters.

Two step responses obtained with the final regulator and estimator values are illustrated in figures 5.1 and 5.2. The large-scale step response in figure 5.1 shows the coarse stages action driving the piezo out of saturation and moving it to the desired reference value. Figure 5.2 shows that the fine stage is very quick to reach a value in its action range. Furthermore, the figure shows that the fine stage compensates the movement of the coarse stage as soon as this approaches the reference.

The equations of the system are perturbed by zero-mean white-gaussian noise on the process and on the measured output. It is assumed that the measurement noise has a greater variance than the process noise. The Kalman filter is supposed to filter this noise from the output. In the simulation zero-mean white noise was applied to the
5 SIMULATION AND COMPARISON TO THE SISO DESIGN

Figure 5.1: 50 µm step response with saturating fine stage actuator

System. As the the sampling time of the noise applied to the system is way lower than the sampling time of the system the noise is not gaussian. The purpose of applying this perturbation rather than applying white noise is to show the reaction of the filter to a low frequency disturbance as e.g. an oscillation of the carrier platform.

Figure 5.3 is a magnification of the step response in figure 5.1 when the coarse stage has finally reached the reference value. The dotted line shows the reference, which is to be tracked. The above described perturbation to the system and its addition to the reference is shown by the dash-dotted line. Further, the estimation, which provides the feedback for the controller is illustrated by the solid line. It detects the perturbation as soon as it occurs and therefore the controller is able to reject it from the output. This rejection is indicated by the fact that the position (dotted-line) does not follow the noised reference, but returns to the “real” reference very quick. In addition, the Kalman filter’s tuning showed that a too aggressive discrete Extended Kalman filter destabilized the response of the fine stage. That occurs when the order-of-magnitude of the measurement covariance approaches the value of the process noise covariance.
However, a value of measurement covariance $V$ three orders-of-magnitude larger than the process noise covariance $W$, resolved in the stable solution illustrated in figure 5.3.

![Noise impact](image)

**Figure 5.3:** Impact of process and measurement noise on position and estimated position

The following performance analysis corresponds to the simulation of the tracking of the Kolmogorov signal as it is shown in figure 5.4. The solid line shows the output position of the model which is almost superimposed to the reference signal (dotted line). Further, the coarse stages position is to be seen as the dashed line in the figure. It shows that the coarse stage tracks the signal very well.

The corresponding fine stage action is out of the figures scale and is shown in figure 5.5. This figure shows the fine stage actuator’s metrology and its saturation boundaries. A very important point is that the fine stage never reaches its saturation so that it can always react to a sudden reference change. In the chosen configuration the saturation limit is not reached by a factor of 10. This guarantees the reliability of the achieved performance for similar signals. This performance was achieved by an aggressive tuning of the coarse stage.

The most important performance criteria of the controller is the RMS error between the reference signal and the position. The ESO requires a maximum RMS of 70 nm. Figure 5.6 shows the achieved RMS of approximately 13 nm under the presence of the in figure 5.3 shown “noise” amplitudes. The values illustrated are the absolute values of the tracking error. Further the overall maximum error is marked with an amplitude of 74 nm.
Kolmogorov signal tracking

![Kolmogorov tracking plot](image)

Figure 5.4: Kolmogorov tracking plot

Fine stage action during tracking

![Fine stage action range during Kolmogorov-signal tracking](image)

Figure 5.5: Fine stage action range during Kolmogorov-signal tracking
5.3 Comparison to SISO design

The system already implemented on the controller has been described in section 2.2. The performance achieved by this controller was published in [25] and [26]. For the benchmark, the available data is taken from figure 5.7 and shown in table 5.1.
Table 5.1: Available benchmark data

<table>
<thead>
<tr>
<th>Remaining tracking error</th>
<th>&lt; 0.008 µm RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum instantaneous error</td>
<td>= 0.030 µm</td>
</tr>
<tr>
<td>Maximum piezoelectric expansion</td>
<td>≈ 2.5 µm</td>
</tr>
</tbody>
</table>

The system response shown in figure 5.7 tracks exactly the same Kolmogorov input signal as the one used in section 5.2. The values achieved in the simulation of section 5.2 are shown in table 5.2.

Table 5.2: Simulation results

<table>
<thead>
<tr>
<th>Remaining tracking error</th>
<th>&lt; 0.013 µm RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum instantaneous error</td>
<td>= 0.074 µm</td>
</tr>
<tr>
<td>Maximum piezoelectric expansion</td>
<td>≈ 0.5 µm</td>
</tr>
</tbody>
</table>

The comparison of the figures, as well as the values stated in the tables, show that the coarse stage of the LQG-controlled system is much more active than in the controller installed on the system. Thus, the piezoelectric activity is not as high as in the decoupled SISO design. The preparation for the simulation results and the tuning in section 4.3 have shown, that the activity of the fine stage is largely affected by the reciprocal activity of the coarse stage.

The tracking performance of the LQG-controller is, according to the tables, not as good as the decoupled-SISO systems performance. As both systems can be tuned to serve different purposes, e.g. speed, settling time or tracking error, a comparison is not of much use. The stability of the controller used to achieve the results in figure 5.7 is not considered. Further, the impact of the real influences, e.g. static friction, are not included in the simulated results and therefore, too many differences are present to achieve a significant comparison.

No tuning of the controller was able to achieve a simulated RMS smaller than 12.5 µm. This is very likely due to the fact that the internal controller of the piezoelectric was not tuned to achieve good tracking performances. The bandwidth of the identified model is almost 25% below the real capability of the piezoelectric actuator. So many effects will contribute to the performance of the developed controller on the real system. Physical effects not considered in the model, e.g. static friction, will decrease the performance of the coarse-stage and the tuning of the piezoelectric will increase the performance of the fine stage. As the action range of the piezoelectric is way below its possibilities it will be able to compensate for a less performant coarse stage. Therefore, the tuning after the implementation on the real system will show the real performance and capabilities of the developed controller.
Chapter 6

Implementation

Within this chapter, the hardware configuration used in the context of the implementation is illustrated. Afterwards, the software implementation is introduced. A final discussion about the results of the implementation procedure concludes this chapter.

6.1 Hardware configuration

The hardware available for the implementation of the algorithm was specified and accepted by the ESO. It contains a Local Controller Unit (LCU), amplifiers for the voltage output signal of the system, the in Chapter 3 described actuators as well as a measurement system. The links between these elements are shown in Figure 6.1 and explained below.

The LCU provides the output voltage of the control algorithm. This digital output is converted to an analogous signal in a D/A converter (MX58). For the coarse stage it is linearly amplified with a voltage amplifier before it is applied to the coils of the UltraMotion NEMA 23 stepper motor. Over the leadscrew, the motors output is transformed to the translational output of the coarse stage $y_c$. The LCU output voltages to the fine stage enter the local PI controller (E-509). The output of the E-509 compensates for the hysteresis of the piezoelectric and is fed to the PI amplifiers (PI

![Hardware Setup](image-url)
E-505). The amplified output is applied to the actuators which provide the fine stage position $y_f$.

The combined position $y = y_f + y_c$ is sensed by an Agilent interferometer positioning system. A laser head (Agilent 5517A) emits a laser beam which enters the measurement optics, the interferometer and the retroreflector. The measurement is picked up in a dedicated receiver (Agilent 10780C,F) and processed on a VMEbus laser axis board (Agilent 10897B). This measurement scheme is illustrated in figure 6.2. The resolution achieved with this configuration is approximately 1.5 nm [27].

![Figure 6.2: Agilent Measurement Setup](image)

As the cat’s eye is not present at this moment, a dummy is directly mounted on top of the blade-guiding-system. Onto this dummy, at a position equivalent to the one in the cat’s eye, the piezoelectric actuator is fixed. Figure 6.3 shows an image of the DDL in the test environment.

![Figure 6.3: The DDL in the setup environment](image)
6.2 Software implementation

A real time software of the ESO called TAC, using VxWorks, is installed on the LCU. The TAC-software provides a block oriented programming environment. The implemented algorithms can be controlled over a graphical user interface and are to be programmed in C.

The, in chapter 4 developed, discrete control scheme has been translated into C to suit this requirement. In figure 6.4 a principle scheme of the implemented C-code is illustrated. It shows the input and output to the function and its initialization.

![Diagram of algorithm sequence](image)

Figure 6.4: Principal sequence of the implemented algorithm

6.3 Results and discussion

After implementing the code in the complex programming environment the initialization of the algorithm ran without problems. When implementing the main function the software was not able to handle the code. This was shown by the slow processing of the slave device which took several seconds to respond. In order to analyze this error, the code was step-by-step commented and the critical point seems to be a 8-by-8 matrix multiplication in the Kalman filter a posteriori update. It seems that all flops up to this point add to the problem, either the processor speed or the memory size is not sufficient.

According to the lack of time, no further implementation tests could have been run. If additional tests are taken they should consider the verification of the block dd1EpflAlgo in the TAC environment as well as the change of the datatype in the
code to single.

The resulting speed or memory error of the controller verifies the statement in section 2.3.4, whereas many problems occurred in the implementation of the modern control approaches due to the high-order of the algorithms.
Chapter 7

Conclusion

This thesis aimed to develop a linear quadratic gaussian regulator for improving the performance of astrometric observations. The DDL to which the controller is applied achieves high-nanometer-accuracy over a scale of many millimeters. This accuracy is required to enhance the reach and improve the precision of earthbound observation.

Section 2.1 of this thesis showed the origin of the high-precision requirement and gave a short introduction to the theoretical background of the basic principles of astrometric research. The following section 2.2 provided the hardware setup and the controller layout, that set the benchmark for the development of the subsequent chapters. The currently implemented controller is a manually tuned system that does not rely on a mathematical method. Hence, the development of a LQG controller was proposed in order to find an optimal solution to the tracking problem. Furthermore, the LQG design is supposed to increase the performance and to consider different requirements in its design, e.g. the power consumption. Section 2.3 provided the necessary fundamentals to develop this controller and showed the common application of dual-stage structures in HDD position control as well as the methods used to control them. It became apparent that, compared to a regular dual-stage decoupled SISO control algorithm, the development of a multivariable controller with state estimator is very complex. The mathematics of the optimization and estimation are not trivial and the high-order of the multivariable controller is prone to creating problems at the implementation stage. Further, it showed that, as the DISO structure is overactuated, to achieve full feedback information, a measurement of one of the two stages positions is necessary. However, this measurement is not available and therefore a very precise model is required.

Therefore, a very thorough modeling was completed in Chapter 3. The nonlinear equations of the coarse stage were linearized using the Park transformation with nonlinear feedback. An evolutionary algorithm provided an accurate set of model parameters to these equations. The identification of the fine stage was completed with a standard MATLAB® identification algorithm and the two models were considered to create the overall system model. The analysis of the system properties already indicated that the system is not observable and that a special emphasis on the estimator design is required.

In Chapter 4, the multivariable state controller and the state estimator were developed and discretized. The unobservability of the system resulted in the state estimator
being unable to estimate all states perfectly. Hence, a bias originated on the unobservable states. As these states are stable, their estimates can still be used. The discretization of the Kalman filter showed that the use of a standard Euler step forward discretization did not provide a good enough a priori estimation and hence a RK4-method was introduced. Further, the level of uncertainty in the noise assumption to the Kalman filter showed that the tuning of the filter had to be fulfilled on the real system. The design of the regulator itself was straightforward and an intense tuning was completed.

The system has been simulated in Chapter 5. However, the results did not reach the measured values of the decoupled SISO design, but they still provided very good results in comparison to the ESO requirements. Adding the Kalman filter to the system is able to destabilize the controller. Hence, the tuning on the real system, under the influence of real noise, becomes more important as the offline is highly model dependant.

The final implementation in Chapter 6 resulted in a speed or memory problem of the LCU. Most likely this problem was due to the high-order of the state feedback controller and the large number of operations taking place in the algorithm. This result verified the statements found in several papers, e.g. [1], that the high-order of the multivariable controller leads to problems at the implementation stage. As this hardware has been specified and accepted by the ESO, it is unlikely that it will be changed. Hence, this paper proposes that the implementation of the decoupled SISO structure, that fulfills the requirements very well, is maintained. In terms of the future application the implementation of the decoupled PI- and I-controller to the structure would also be a good alternative in terms of maintenance and complexity.

Even, if the implementation of the controller on the hardware did not provide satisfying results, the thorough methodical approach in this thesis increased the global knowledge of the system. A very accurate model of the coarse-stage has been obtained and a deep insight on the methodology of LQG control could be achieved. During this thesis, the PRIMA consortium changed the hardware of the coarse stage from voltage to current input. This change affects the properties of the system completely. It reduces the order of the controller and provides an observable system. It would be very interesting to design and implement a controller and a fully operable observer on this system to compare its performance to the decoupled SISO design.
Bibliography


[16] **Hespanha, Joao P.** *Undergraduate Lecture Notes on LQG/LQR controller design*. Lecture notes, University of California, Santa Barbara, May 9, 2006.


