



TECHNISCHE UNIVERSITÄT ILMENAU

Dissertation

**Zur Erlangung des akademischen Grades
Doktoringenieur (Dr.-Ing.)**

**Robust Real-Time Control of an
Adaptive Optics System**

von

M.Sc. Steffen Mauch

geboren am 15.02.1987 in Sigmaringen

vorgelegt der Fakultät für Informatik und Automatisierung
der Technischen Universität Ilmenau.

Tag der Einreichung: 29. April 2016
Tag der Verteidigung: 29. August 2016

Erstgutachter: Univ.-Prof. Dr.-Ing. Johann Reger, TU Ilmenau
Zweitgutachter: Univ.-Prof. Dipl.-Ing. Dr. techn. Martin Horn, TU Graz
Drittgutachter: Dr.-Ing. Erik Beckert, Fraunhofer IOF Jena

urn:nbn:de:gbv:ilm1-2016000431

*This thesis is dedicated to my family, especially to my parents
for their never-ending support and encouragement;
even in the toughest moments.*

Acknowledgements

First and foremost, I would like to thank my parents for their love and endorsement throughout my studies and life; they have always supported me as best as they could.

Furthermore, my sister deserves my wholehearted thanks together with her lovely little girl Malini. She is always so adorable and my little sunshine.

I would like to express my special thanks to my supervisor, Univ.-Prof. Dr.-Ing. Johann Reger, for his guidance and support throughout this study, especially for his confidence in me after finishing my studies and offering me the Ph.D. position.

Additionally, I want to say thank you to every member of the Control Engineering Group at the Ilmenau University during my time there. It was a pleasure having you around and working with you; especially many thanks to Alexander Barth for supporting and executing several experiments in the final stage.

I sincerely thank the Adaptive Optics Group of the Fraunhofer Institute for Applied Optics and Precision Engineering (IOF) for providing the optical components and letting me share their knowledge of the calibration of these, especially Dr.-Ing. Erik Beckert for his commitment during the whole time as well as M.Sc. Michael Appelfelder for his support and many long working days in the institute.

To all my friends, thank you for your understanding and encouragement in my moments of doubt. Especially to Katha for her endless motivation and support throughout the whole time and the unforgettable adventures during the time spent with you.

Messkirch, 09/2016

Abstract

This research contributes to the understanding of the limitations when designing a robust control real-time system for Adaptive Optics (AO). One part of the research is a new method regarding the evaluation of a Shack-Hartmann wavefront sensor (SHWFS) to enhance the overall performance. The method is presented based on the application of a Field Programmable Gate Array (FPGA) using Connected Component Labeling (CCL) for blob detection. The FPGA has been utilized since the resulting delay is crucial for the general AO performance. In this regard, the FPGA may accelerate the evaluation largely by its parallelism. The developed algorithm does not rely on a fixed assignment of the camera sensor area to the lenslet array to maximize the dynamic range.

In extension to the SHWFS evaluation, a new rapid control prototyping (RCP) system based on hard real-time RTAI-patched Linux kernel has been developed. This system includes the required hardware e.g. the analog output cards and FPGA based frame-grabber. Based upon a Simulink model, accelerated C/C++ code is automatically generated which uses the available parallel features of the processor. A continuative contribution is the application of a robust control scheme using \mathcal{H}_∞ methods for designing a controller while considering uncertainty of the identified model. For synthesizing the controller, a special optimization technique called non-smooth μ -synthesis is utilized which minimizes the \mathcal{H}_∞ norm while coping with pre-specified controller schemes. Depending on the pre-specified controller scheme, the resulting controller can be computationally costly but the RCP approach is designed to cope with the problem. Based on simulations and according to experiments, the validity of the identified models of the AO setup is assured. At the same time, the enhanced performance of the new RCP setup is demonstrated.

Kurzfassung

Die wissenschaftliche Arbeit trägt maßgeblich zum Verständnis der gängigen Limitierungen bei robusten echtzeit-fähigen Regelungssystemen für Adaptiv Optische (AO) Systeme bei. Ein wesentlicher Teil der Arbeit befasst sich mit einer neuartigen Methode der Auswertung eines Shack-Hartmann Wellenfrontsensors (SHWFS). Die Methode basiert auf der Anwendung eines Field Programmable Gate Arrays (FPGA) zur Auswertung des SHWFS.

Die zu Grunde liegende Methode ist ein Resultat der Graphentheorie zur Erkennung zusammenhängender Bildbereiche. Der Einsatz eines FPGA ermöglicht hierbei, dass die resultierende Verzögerung durch die Auswertung des SHWFS auf ein Minimum reduziert wird. Hinzu kommt, dass die neuartige Auswertungsmethode den dynamischen Bereich des Wellenfrontsensors gegenüber dem üblichen Verfahren erweitert, da für die Methode keine feste Zuordnung der Spots zu dem Linsenarray notwendig ist.

Zusätzlich zu dem neuartigen Verfahren für die Auswertung wurde ein Rapid Control Prototyping (RCP) System entworfen, welches auf einem echtzeit-fähigen Linux Kernel basiert. Die Echtzeitfähigkeit wird durch die Verwendung des Real-Time Application Interface for Linux (RTAI) erreicht. Der Entwurf des RCP Systems umfasst die Entwicklung spezieller Hardware wie beispielsweise eine analoge Ausgangskarte und der PCIe FPGA Framegrabber. Aus einem Simulink Modell wird automatisch C/C++ Quellcode generiert. Dieser generierte Code nutzt die vorhandenen parallelen Erweiterungen des Prozessors zur Beschleunigung der vorkommenden Berechnungen. Basierend auf diesem System wurde ein robustes Regelungsverfahren angewendet, welches auf der \mathcal{H}_∞ Entwurfsmethodik basiert. Das Entwurfverfahren des Reglers (non-smooth μ Synthese) berücksichtigt die vorhandene Unsicherheit der identifizierten Modelle bereits während der Synthese. Das Verfahren ermöglicht die \mathcal{H}_∞ Norm des geschlossenen Regelkreises zu minimieren, wobei die Regler-Struktur vorgegeben werden kann. Basierend auf verschiedenen Simulationen und experimentellen Versuchen wurde die Gültigkeit der identifizierten Modelle des AO Systems nachgewiesen. Zudem wird gezeigt, dass das entworfene RCP System deutlich leistungsfähiger als vergleichbare Systeme ist und somit eine deutlich verbesserte Performance aufweist.

Contents

Glossary	iii
List of Symbols	v
1 Preliminaries	1
1.1 Introduction to Adaptive Optics (AO)	1
1.2 Challenges of AO	3
1.3 Contribution of this Thesis	5
2 Adaptive Optics	7
2.1 Deformable Mirror	9
2.2 Wavefront Sensing	12
2.2.1 Shack-Hartmann wavefront sensor (SHWFS)	13
2.2.2 Curvature wavefront sensor	15
2.2.3 Pyramid wavefront sensor	16
2.2.4 Shearing interferometer	17
2.2.5 Holographic wavefront Sensor	18
2.3 Experimental Optical Setup	20
3 Real-Time Wavefront Measurement Algorithm	23
3.1 Wavefront Reconstruction *	25
3.2 Conventional Approach	32
3.3 Contributed Algorithms for Evaluation of SHWFS	34
3.3.1 Overview over SHWFS evaluation approaches	34
3.3.2 Connected component labeling (CCL)	35
3.3.3 Straight line centroid ordering	39
3.3.4 Spiral spot pattern adaption	42
3.3.5 Adaptive thresholding	47
3.3.6 Repositioning of centroids	49
3.4 FPGA Algorithm Implementation	52
3.4.1 CCL for SHWFS's	52
3.4.2 Straight line centroid ordering	55
3.4.3 Modified spiral algorithm	60
3.4.4 Real-time adaptive thresholding	64
3.4.5 Reordering of centroids by similarity	65
4 Robust Optimal Control	67
4.1 \mathcal{H}_∞ Framework	69
4.1.1 \mathcal{H}_∞ preliminaries	69
4.1.2 Handling uncertainty	75
4.1.3 Structured singular value (μ)	77
4.1.4 \mathcal{H}_∞ optimal controller design	80
4.1.5 Controller design via non-smooth μ -synthesis	83
4.2 Modeling	87
4.2.1 Tip-Tilt mirror	88

4.2.2	Deformable mirror	91
4.2.3	Wavefront sensor	97
4.2.4	Delay handling	101
4.2.5	Real-time system	103
4.2.6	Weighting matrices	104
5	Experimental Setup	107
5.1	System Setup	110
5.1.1	Hardware platform	111
5.1.2	Software platform	114
5.2	Experimental Results	119
5.2.1	Tip-tilt control	119
5.2.2	Deformable mirror control	124
6	Conclusion	141
6.1	Overview	141
6.2	Future Prospects	143
	List of Figures	145
	List of Tables	149
	Listings	151
A	Appendix	153
A.1	Fraunhofer Diffraction Pattern	153
A.2	Algebraic Riccati Equation and Inequality	154
A.3	Bounded Real Lemma	155
A.4	Bisection	156
A.5	D/K-Iteration	156
A.6	Latency/Performance Measurements	157
A.7	Matlab/Simulink Scripts/Models	162
A.8	Scanning Vibrometer	165
A.9	Imagine Optics HASO TM 3 Fast	167
A.10	Simulation and Experimental Data	169
A.11	List of Own Publications	171
B	Bibliography	173

Glossary

a.u.	Arbitrary Unit
ADC	Analog-Digital Converter
Adeos	Adaptive Domain Environment for Operating Systems
AIF	Actuator Influence Function
AO	Adaptive Optics
API	Application Programming Interface
ARE	Algebraic Riccati Equation
ARI	Algebraic Riccati Inequality
ARX	AutoRegressive with eXogenous inputs
ASCII	American Standard Code for Information Interchange
AVX	Advanced Vector Extensions
AVX2	Advanced Vector Extensions 2
BIBO	Bounded-Input Bounded-Output
BMBF	Bundesministerium für Bildung und Forschung
BMI	Bilinear Matrix Inequality
BRAM	Block Random Access Memory
CBS	Cube BeamSplitter
CCD	Charge-Coupled Device
CCL	Connected Component Labeling
CMOS	Complimentary Metal Oxide Semiconductor
Comedi	Linux Control and Measurement Device Interface
CORDIC	COordinate Rotation DIgital Computer
CPU	Central Processing Unit
DAC	Digital-Analog Converter
DLL	Dynamic Link Library
DM	Deformable Mirror
DM40	Deformable Mirror having 40 Actuators
DMA	Direct Memory Access
dSPACE GmbH	Digital Signal Processing And Control Engineering GmbH
E-ELT	European Extremely Large Telescope
FF	Flip-Flop
FFT	Fast-Fourier Transformation
FIFO	First In First Out
FMA3	Fused Multiply-Add instruction 3
FPGA	Field Programmable Gate Array
GCC	GNU Compiler Collection
HOE	Holographic Optical Elements
HV64	High Voltage Amplifier with 64 channels
HWFS	Holographic Wavefront Sensor
IOF	Institute for Applied Optics and Precision Engineering
IP	Intellectual Property
IQC	Integral Quadratic Constraint

KD OptiMi	KD OptiMi - Kompetenz Dreieck Optische Mikrosysteme
LabVIEW	Laboratory Virtual Instrumentation Engineering Workbench
Laser	Light Amplification by Stimulated Emission of Radiation
LFT	Linear Fractional Transformation
LMI	Linear Matrix Inequality
LTCC	Low Temperature Cofired Ceramics
LTI	Linear Time-Invariant
LUT	Lookup Table
LXRT	module that allows to use RTAI features and its schedulers in user space, both for soft and hard real-time
MATLAB	MATrix LABoratory
MEMS	MicroElectroMechanical Systems
MIMO	Multiple Input Multiple Output
NI	National Instruments
OS	Operating System
OSC	Optical Sciences Center
P-WFS	Pyramid Wavefront Sensor
PC	Personal Computer
PCIE	Peripheral Component Interconnect Express
PXI	PCI eXtensions for Instrumentation
RCP	Rapid Control Prototyping
RMS	Root-Mean Square
RTAC	Real-Time Atmospheric Compensator
RTAI	Real-Time Application Interface
RTHAL	Real-Time Hardware Abstraction Layer
RTOS	Real-Time Operating System
SHWFS	Shack-Hartmann Wavefront Sensor
SIMD	Single Instruction Multiple Data
SISO	Single Input Single Output
SLM	Spatial Light Modulator
SNR	Signal-to-Noise Ratio
SPI	Serial Peripheral Interface
SSE	Streaming SIMD Extensions
SSV	Structured Singular Value
SVD	Singular Value Decomposition
TCP/IP	Transmission Control Protocol/Internet Protocol
TLP	Transaction Layer Packet
UART	Universal Asynchronous Receiver/Transmitter
UDP	User Datagram Protocol
USB	Universal Serial Bus
VGA	Video Graphics Array
VHDCI	Very-High-Density Cable Interconnect
VHDL	VHSIC Hardware Description Language
VHSIC	Very High Speed Integrated Circuit
WFS	Wavefront Sensor
XST	Xilinx Synthesis Technology

List of Symbols

\mathbb{C}	Complex Number
\mathbb{C}^-	Open Left-Half Complex Plane
$\overline{\mathbb{C}^+}$	Closed Right-Half Complex Plane
Δ	Scalar Uncertainty
$\mathbf{\Delta}$	Class of Uncertainties Δ
\mathcal{I}_n	$n \times n$ Identity Matrix
$G(s)$	Transfer matrix defined in the Laplace Domain
Γ	Control Matrix
\mathcal{H}_2	\mathcal{H}_2 Space
$\ \cdot\ _2$	\mathcal{H}_2 Norm of \cdot
\mathcal{H}_∞	\mathcal{H}_∞ Space
$\ \cdot\ _\infty$	\mathcal{H}_∞ Norm of \cdot
ℓ_2	ℓ_2 Space
\mathcal{F}_l	Lower Linear Fractional Transformation
\mathcal{F}_u	Upper Linear Fractional Transformation
ℓ_∞	ℓ_∞ Space
$\mu(\cdot)$	Structured Singular Value of \cdot
$O(\cdot)$	Big O Notation of Time Complexity
\mathbf{p}	decision parameter
$\phi(\vec{r})$	Continuous Wavefront Phase at \vec{r}
\mathbb{R}	Real Number
\mathcal{RH}_2	\mathcal{RH}_2 Space
\mathcal{RH}_∞	\mathcal{RH}_∞ Space
$\sigma(\cdot)$	Singular Value
$\bar{\sigma}(\cdot)$	Maximum Singular Value
$\underline{\sigma}(\cdot)$	Minimum Singular Value
τ	Time-Delay
Θ_v	Column Vector of the Wavefront Slopes

1 Preliminaries

The most exciting phrase to hear in science, the one that heralds the most discoveries, is not “Eureka!” (I found it!) but “That’s funny ... “.

ISAAC ASIMOV

The chapters of this thesis could be read individually; whenever conjunctions are present or preliminaries are required, it is marked by referencing the corresponding chapter or specific section.

Each chapter is constructed in such a way that the chapter can be read one section after another. When a section or subsection is not absolutely necessary for the understanding e.g. when discussing a special aspect in detail, the heading is marked with a star at the end.

As the topic of this thesis has been arranged for several years of intensive research, several excerpts of the findings have already been published at conferences or as journal paper (see Appendix A.11).

1.1 Introduction to Adaptive Optics (AO)

Adaptive optics (AO) is a technology which is used to enhance the performance of an optical system by adjusting the wavefront [111, 6, 40, 114]. One of the first applications relying on AO has been ground-based astronomical telescopes [48, 115, 52]. For ground-based astronomical telescopes the objective is to compensate the occurring disturbances in the incoming wavefront.

The key disturbance in ground-based astronomical telescopes is the earth atmosphere [6, 40, 114] which leads to blurring, twinkling and scintillation [114, 40] of astronomical objects due the different optical refractive index [51] and gust of wind [40]. Such disturbances blur the image and thus degrades the resolution limit when observing astronomical objects. This phenomenon is called astronomical seeing; the Kolmogorov model describes the turbulence quite well [107, 51, 6].

The key component of an adaptive optics system is the deformable mirror (DM) which is responsible for the optical correction by manipulating the incidental wavefront. The general theory of operation of an AO system is depicted in Fig. 1.1 whereas Fig. 1.1 depicts a closed-loop AO setup. To be able to perform a correction of the wavefront, the wavefront has to be measured; more precisely, the phase of the wavefront has to be gathered with an appropriate device (wavefront sensor). Ideally the wavefront sensor is able to measure the shape of the wavefront faster than the occurring disturbance, as required by the Nyquist-Shannon sampling theorem. Several

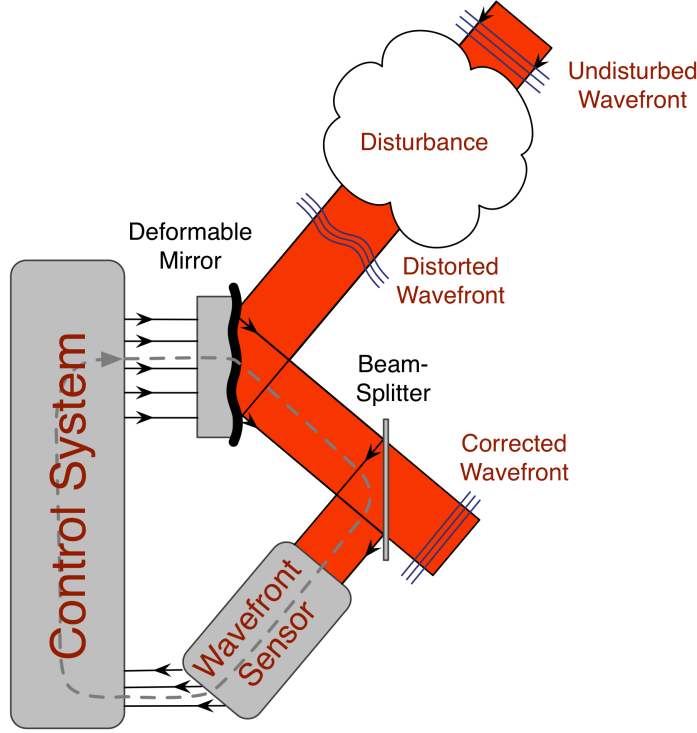


Figure 1.1: General adaptive optics (AO) principle

different concepts regarding wavefront sensing have been developed up to now. The Shack-Hartmann wavefront sensor (SHWFS) is the most common wavefront sensor for AO application being used for most applications (see Sec. 2.2). The wavefront sensor delivers the achieved information to the controller (control system), in general a high-speed computer which then calculates the required deformation of the DM. The controller determines the appropriate signals (closed-loop mode) to compensate for the disturbance. Except from correcting atmospheric disturbances, endless other applications for ground-based astronomical telescopes have been demonstrated in the past which rely on AO [40, 114].

The simplest form of adaptive optics is the tip-tilt correction, where solely the tilts of the wavefront are corrected. Tip-tilt mirrors could be classified as segmented mirrors (see Sec. 2.1). These kind of mirrors have only one segment which can tip and tilt instead of multiple segments that can tip and tilt independently. They offer two degrees of freedom (tip-tilt) which are independent of each other. Tip-tilt mirrors have, due to their simplicity, a large stroke and therefore can correct large tip-tilt aberrations. Most AO systems use a separate tip-tilt mirror in addition to the DM such that higher order correction is not unnecessarily limited due to large tip-tilt aberrations. Higher order aberrations have in general lower amplitudes than low order e.g. tip-tilt aberrations. Nevertheless, only one DM is drawn in Fig. 1.1 as it shall only represent the fundamental concept of AO.

1.2 Challenges of AO

The troublesomeness when drafting an AO system will shortly be discussed in the following to catch a feeling for the challenges of an AO system.

Foremost, the design and selection of the individual components for an AO system requires interdisciplinary skills such as optical and system theoretical as well as engineering knowledge in general. For example, the required focal length of the lenses to adapt the different apertures to each other have to be calculated. For simulation of the optical setup, Zemax is commonly used which is an optical design program. For the work presented in this thesis, some components have already been obtained such as the DM and the SHWFS. These components could not have been replaced for further optimization of the optical setup. The performance of the DM is essential for the accuracy when manipulating the distorted wavefront. Thus the DM must offer the required degree of freedom to shape the wavefront for the corresponding spatial distortions [113]. The surface of the DM itself must not have huge surface irregularities as this degrades the performance further. Additionally, the speed of the DM, coupling between the individual actuators and maximum achievable deflection will highly influence the performance as well.

Depending on the application which the AO system is used for, different requirements are prevalent for the required bandwidth of the closed-loop. For typical turbulence and thus wind condition, the necessary bandwidth for atmospheric compensation is in the order of 100 Hz for visible light [34]. The required correction bandwidth is given by the Greenwood frequency f_G [33] which is a measure of the variation over time of the Kolmogorov atmospheric turbulence. As the bandwidth scales with the wavelength $\lambda^{-6/5}$, the requirements decrease when moving towards the infrared instead the visible region. As a rule of thumb, the bandwidth of the closed-loop system should be at least ten times faster than the Greenwood frequency [114, page 10]. For laser link application between a ground station and low-earth orbit and geostationary satellite, the demanded bandwidth is even higher as low elevation angles occur. The necessary greenwood frequency f_G can result to > 1 kHz under these conditions [50].

The adjustment of the optical setup itself is very crucial for reproducibility as it can decrease the maximum performance in terms of static induced aberrations. Depending on the selected method for sensing the wavefront, the adjustment can limit the sensing dramatically as these static aberrations can hinder a correct measurement. Artifacts such as diffraction or reflection can make the measurement even more complicated. In fact the measurement of the wavefront with an appropriate speed, while having low latency and a deterministic behavior regarding sampling, is challenging. The software being offered by the manufacturer of the devices for wavefront sensing is generally closed-source. Furthermore the software often neglects the requirements which are imposed by closed-loop operation such as being deterministic and minimizing the latency during evaluation. Due to the closed-source it is hard to judge whether the

algorithm being used is suitable for the desired application.

The design of the controller itself is also somehow challenging too as the AO system has in general a multiple input multiple output (MIMO) characteristic. The use of empirical methods for designing the controller is quite common in literature as well as for real applications. Nevertheless empirical methods do not allow easy robustification against uncertainties and neglect mostly the MIMO characteristic. When testing new approaches, as much as possible elements of uncertainty must be eliminated to be successfully in reasonable time. Therefore a rapid control prototyping (RCP) system is highly desirable which should have a direct link to MATrix LABoratory (MATLAB) and Simulink for easy testing.

Ultimately from the perspective of control, the foremost goal is to maximize the performance of a given system. Thus in this case achieving superb performance under the given circumstances e.g. the wavefront sensor, control system and DM. As control engineer it appears somehow as legitimate to take a given setup and design a controller for the DM to maximize the performance without understanding the imposed limitations of the individual involved components. Nevertheless, the overall result would not be as good as if the whole setup is assessed for optimization as the individual components are also crucial for the overall performance. Therefore this work also focuses on identifying the bottlenecks and removing them as far as it is possible under the fixed imposed limitations.

1.3 Contribution of this Thesis

- ▶ Development of new evaluation methods for an SHWFS, whereas the algorithms do not rely on a fixed correlation between the area of the image sensor and the lenslets itself. Thus, the centroids of the spots are not required to stay inside the given area of the sub-aperture.
- ▶ Implementation and validation of the straight-line segmentation approach and the modified spiral method to sort the centroids in a matrix using a field programmable gate array (FPGA) for low-latency SHWFS evaluation in real-time.
- ▶ Conceptual design, building and verification of an RCP approach based on hard real-time Linux using real-time application interface (RTAI) and Simulink. Furthermore, design of a peripheral component interconnect express (PCIe) FPGA card including developing the individual periphery, e.g. analog-digital converter (ADC), digital-analog converter (DAC) boards. Programming a driver for the corresponding Linux kernel to integrate the PCIe FPGA into Linux Control and Measurement Device Interface (comedi).
- ▶ Synthesis of a structured fixed-order MIMO \mathcal{H}_∞ controller which stabilizes the closed-loop even under presence of uncertainties. The controller is able to control the coupled deformable mirror based on the non-smooth μ -synthesis approach. Furthermore, design of an \mathcal{H}_∞ optimal PI-controller while considering uncertainties and preserving the MIMO model description.
- ▶ Confirmation of the simulation results which have been determined for the closed-loop using MATLAB and Simulink. Therefore, matchable experiments have been performed with the experimental setup to validate the simulation by experimental results.

2 Adaptive Optics

Many people said to me – partly as a joke but also as a challenge – that the laser was “a solution looking for a problem“.

THEODORE MAIMAN, INVENTOR OF THE FIRST WORKING LASER, (1960s)

The first concept of adaptive optics (AO) was proposed in 1953 by the astronomer Horace Babcock [20]. The forthcoming technologies of this time had not yet been ready for AO. Between the 1960's and 1970's, aerospace and military institutions built the first serious AO system. Since then, many research groups have worked in the field of adaptive optics. Early theoretical work regarding limitations and capabilities of AO systems has been done, e.g. by Freeman Dyson, François Roddier and John Hardy.

Today AO is also used for light amplification by stimulated emission of radiation (laser) communication systems [126, 94, 112, 55], in microscopy [17, 3], retinal imaging systems [72, 97, 58] and many more to name but a few. AO can ensure that the turbulence of the earth's atmosphere has almost no impact on the distortion of images, gathered at the best places in the world for astronomy, e.g. at the Keck Observatory in Hawaii (United States of America) or the Observatorio del Teide in Teneriffa (Spain).

Turbulence causes stars to twinkle so much that astronomers get frustrated because these turbulences blur out the finest details of an observed star or the cosmos itself. These aberrations can only be avoided by observing directly from space which avoids the blurring effect from the atmosphere. The disadvantages are the high costs of space telescopes compared to ground-based ones and limited space for space telescopes. Furthermore, repairing space telescopes, remember e.g. repairing the Hubble telescope in 1993, is very difficult, expensive and requires a lot of preparation time. Furthermore, the acceleration during take-off is extremely high which makes designing the individual parts of a space telescope more complicated. Clearly, they must resist the resulting forces without any damage. Another aspect is that special care regarding cosmic radiation [122, 2] for electronics in space has to be taken and in some cases protecting metal plates have to be built-in which significantly increases the weight of the system. On the other hand, energy consumption must be kept as low as possible so that the power from solar cells is sufficient.

The atmospheric turbulence is yielded by e.g. different temperature layers, differing wind velocities or in general, by different refraction indexes [39, 20]. These effects are called astronomical seeing which results in blurring and twinkling of astronomical objects, such as stars. A description of the wavefront aberrations influenced by the atmosphere is given by the Kolmogorov model [39, 114]. A variety of experimental

measurements support this model.

In most cases, the diameter of the seeing disc (point spread function for imaging through the atmosphere) is used as a seeing measurement. This point spread function diameter is a reference to the best possible angular resolution that can be reached by an optical telescope using a long exposition. This corresponds to the diameter of the blurred blob, seen when observing a point-like star through the atmosphere. Under best conditions (high-altitude observatories at e.g. Mauna Kea or La Palma) a seeing disk diameter of ≈ 0.4 arcseconds is achievable.

The utilization for laser material processing applications is another example (see Fig. 2.1). The main focus in this context is the compensation of thermal lenses [18, 15] and focus adjustment due to thick materials; thus, defocus readjustment. In the case

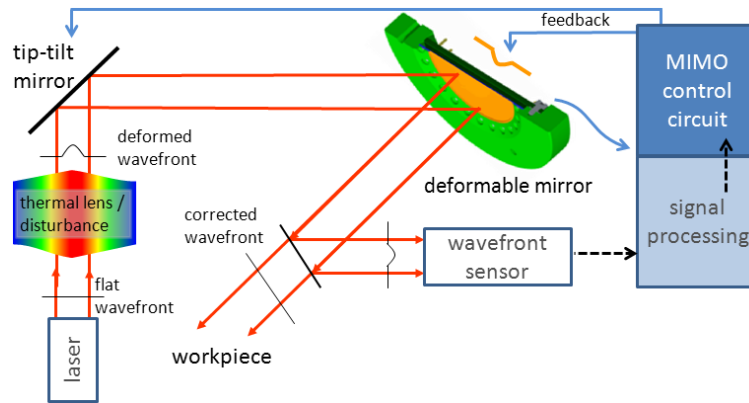


Figure 2.1: Adaptive optics principle for laser material processing [66]

of material processing and high-power lasers, thermal lenses and aberrations/disturbances from laser itself or the optical components [18, 15, 42] degrade the wavefront. The characteristics of the individual components are maintained, but of course the deformable mirror needs to be able to handle the high-power laser beam. The underlying principles of wavefront manipulation and correction remain the same.

The Gaussian beam shape is not always ideal for the application, e.g. an intensity distribution of flat tops (circular or square) or even doughnut shapes [15, Figure 2.18] can be more suitable. The result is likely better machining performance [15, 42]. If e.g. micro structuring should be carried out, even more complex distributions of intensity might be beneficial.

In the following, no focusing on a specific field of application such as laser material processing or atmospheric disturbances compensation takes place because every field of application has the control of at least one deformable mirror in common. Additionally, the wavefront that shall be corrected has to be measured in an appropriate manner.

2.1 Deformable Mirror

Deformable mirrors (DMs) are mirrors whose surface can be deformed in a predefined manner. DMs are commonly used to control and correct optical aberrations while the wavefront is measured by an appropriate sensor in real-time. The change of the DM should be much faster than the dynamics of the occurring aberrations in the optical setup, to minimize the required time to correct a given aberration.

The degree of freedom of a DM is usually given by the number of actuators; even a tip-tilt mirror is a DM but with only two degrees of freedom (tip and tilt).

The main parameters of a DM are:

- ▶ Number of single controllable actuators; indirectly determines the space and thus, the degree of freedom of the disturbance, which the DM can correct.
- ▶ Actuator pitch; distance between actuator centers which define the spatial resolution of the DM.
- ▶ Actuator stroke; maximum possible displacement, typically in positive and negative direction from a common zero joint position; also influences the magnitude of disturbance which can be compensated for.
- ▶ Actuator coupling; movement of one actuator will displace/influence its neighbor; thus affects the complexity of the control problem. This determines whether the control problem is a single input single output (SISO) or multiple input multiple output (MIMO) problem.
- ▶ Response time; measures how quickly the actuator of the DM reacts to a control signal; thus, influences the resulting delay/latency.
- ▶ Hysteresis and creep; non-linear actuation effect which deteriorates the precision of the response of the DM; strongly depends on the commonly used actuators; significantly limits the possibility of open-loop control.

Many of the mentioned parameters of a DM cannot be enhanced without degrading other parameters. For example requiring a large stroke of the actuator with a small actuator pitch will lead to higher actuator coupling when having a continuous DM. A variety of different DM concepts have been developed over the time [62], e.g. segmented mirror, continuous faceplate [90], unimorph/bimorph mirrors, et cetera. The

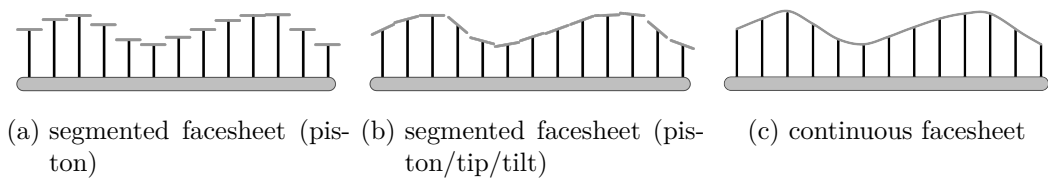


Figure 2.2: main characteristic of different DM concepts

main distinctive concepts for DMs are shown in Fig. 2.2. Segmented facesheet piston

actuators (Fig. 2.2a) allow a very high actuator density. The actuators cannot influence each other because they are all decoupled through the segmentation. One major disadvantage is that light is getting lost due to the spacing and the created surface is not continuous. When the surface is not only adjustable in the piston but also in tip and tilt (Fig. 2.2b), then the discontinuity is significantly reduced. Nevertheless the result is that the neighbors of each actuator have to be incorporated in the control strategy to approximate a good continuous surface. Incorporating the neighbors leads, from a control perspective, to a MIMO instead of a SISO problem, as multiple actuators have to be changed accordingly.

When using a continuous facesheet (Fig. 2.2c) there is no photon loss any more and a low spatial aliasing is achieved. The influence of one actuator on another can be quite strong, depending on the thickness and the material of the facesheet. Thus, in general, the resulting control problem is a MIMO problem, as the system cannot be separated anymore into individual subsystems of lower order, at least. Under specific circumstances, the problem can be reformulated as a SISO problem by applying approximations and decoupling techniques.

As part of the Bundesministerium für Bildung und Forschung (BMBF) project KD OptiMi - Kompetenz Dreieck Optische Mikrosysteme (KD OptiMi) (see Chap. 5) DMs based on low temperature cofired ceramics (LTCC) have been developed at the Fraunhofer Institute for Applied Optics and Precision Engineering (IOF) [90]. A speciality of the mirrors are their capability of being usable in high power laser applications up to 6.2 kW incident laser power [92]. In contrast, conventional unimorph-type deformable mirrors suffer from thermal lensing [91].

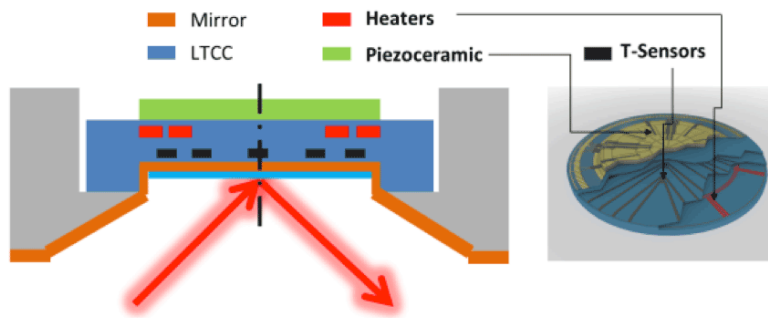


Figure 2.3: Cross-sectional view of the deformable mirror (left side) with buried integrated heaters and temperature sensors. The right side illustrates the rear surface of the mirror with screen-printed actuator structure, integrated heaters and wiring of the temperature sensors. The incoming laser beam is visualized by the red arrows.

The advantage of this approach is the large actuation capability of the thermal actuators that are used to compensate for large amplitudes but reasonably slower variations compared to usual aberrations. Fig. 2.3 shows the set-up of the DM incorporating heater, piezoelectric actuators and temperature sensors. As a mirror substrate, a LTCC membrane is used consistently in a layer-wise setup. One layer

integrates the buried temperature sensors. A second layer integrates buried heaters for temperature control of the mirror assembly. To measure the radial temperature gradient in the membrane, nine sensors are buried; one in the center of the membrane and four on concentric circles with 12 and 24 mm, respectively. Therefore it is possible to detect the temperature of the center that corresponds to the incidental laser beam power. Additionally, the radial temperature profile of the membrane can be detected by the sensors on the additional rings.

During the development of the DMs in the 1970s, (in the case of continuous facesheet DMs) very popular actuators were piezoelectric or electrostrictive materials; capable of high forces, high accuracy, low power dissipation and fast response times. Since 1990, the use of voice coil actuators became popular and afterwards DMs using microelectromechanical systems (MEMS) have been heavily developed due to their compactness, light weightiness and low drive voltage hardware [62]. A very successful company producing DMs based on MEMS is Boston Micromachines Corporation e.g. having a DM consistent of more than 4000 actuators with an active aperture of 25 mm¹ and a stroke of 3.5 μ m.

Regarding the different kinds of actuators, they all differ in the dynamic behavior, their achievable stroke, power consumption and many more. From control perspective, the linearity and in general their model description is very important. Commonly piezoelectric actuators have a hysteresis behavior which leads to non-linear behavior. The occurrence of the hysteresis differs from the material used, the thickness, the voltage applied and several other factors. For each DM, the influence of the hysteresis has to be considered separately.

Nevertheless, the influence of creeping is negligible for the closed-loop mode as it is corrected automatically and the time-scale is much slower compared to the actuator dynamic. The underlying physical effect is as follows: If the operating voltage (control signal) of a piezo actuator is altered, the remaining polarization (piezo gain) continues to change. This manifests in a slow change of position even when the control signal is held constant². In general, the rate of creep decreases logarithmically with time.

Today, depending on the application, these different actuators and DMs are used. At least for high-power and applications dealing with low intensity of the light, the continuous facesheet (Fig. 2.2c) is the preferred concept, as no additional losses occur. Of course, for very large mirrors, e.g. the European Extremely Large Telescope (E-ELT), having a main mirror of 39 m diameter, only a segmented facesheet is reasonable from a practical point of view.

For more detail about the different actuators and kinds of DMs the survey paper of [62] is strongly recommended.

¹ <http://www.bostonmicromachines.com/plus-kilo.htm>

² http://www.piezo.ws/piezoelectric_actuator_tutorial/Piezo_Design_part2.php

2.2 Wavefront Sensing

Wavefront sensing is used to determine the phase of a wavefront for general purpose. In contrast, if the aim is to determine the surface quality of a component with very high precision or even the velocity or acceleration for vibration analysis, laser interferometers are typically used. An interferometer is a technical device based on the interference of waves, typically light waves. The interferometer delivers high precision length measurements but requires a reference source.

In the case of AO when unknown disturbances shall be compensated or only measured, direct measurement of the slope/gradient of the wavefront is sufficient. This is true as no reference source is required when using direct methods. Furthermore, typical wavefront sensors (WFSs) can capture the wavefront at several discrete points. Laser interferometers instead can only measure one point at a time. If multiple points have to be captured in parallel to achieve an in-phase measurement, a separate interferometer is required for each point. Of course, it would be demanding too much if the wavefront is captured continuously instead at discrete points only. But the most important characteristic for control purposes is to capture the wavefront in parallel, called real-time measurement.

Wavefront sensing can be divided into direct and indirect wavefront sensing [39]. Direct wavefront sensing means that the wavefront is measured either by zonal or modal approaches, whereas indirect wavefront sensing evaluates the effect of a related parameter, in most cases the distribution of intensity. Two methods of indirect wavefront sensing are aperture tagging and wavefront deconvolution, but none of these are suitable for AO [39]. Therefore only direct wavefront sensing methods are part of this overview.

Till nowadays, the two most commonly used major types of wavefront gradient sensors [39] are the Shack-Hartmann wavefront sensor (SHWFS) and the shearing interferometer; both being zonal approaches. The shearing interferometer takes a different approach compared to the SHWFS. The SHWFS spatially segments the wavefront, which is explained in detail in Sec. 2.2.1, whereas the shearing interferometer duplicates the wavefront to be measured [39]. Additionally, a curvature WFS has also been developed [95] to measure the curvature of the wavefront; other approaches are the pyramid WFS [88] and the holographic WFS [30]. For each of the wavefront sensing approaches, a lot of variations have been presented in the last years, each being suited for special applications.

The SHWFS is the most commonly used WFS due to its simplicity. The characteristics of the SHWFS are sufficient for most applications. Compared to other wavefront sensing techniques, the SHWFS provides the advantages of a broad range, being relatively fast, can be technically realized without much difficulties and is also moderately expensive. In recent years, nevertheless, wavefront sensing, using an SHWFS has been subject to rapidly increasing demands for performance and reduction of latency. More precisely, the imposed demands in terms of sensitivity, resolution and required speed

of the underlying camera of the SHWFS have continuously grown. Furthermore, the reduction of the latency that results from the evaluation of the camera image has received significant attention [14, 32, 69].

Of course, considering special assignments, other wavefront sensing techniques have beneficial properties which the SHWFS does not have, e.g. in extremely low-light situations or when requiring sensor rates (frame rate) up to 1 MHz. These advantages are usually bound to other disadvantages such as cost or practical/technical limitations. Nowadays, the SHWFS is the all-rounder of the wavefront sensing devices, suitable for most challenges and not being determined to one specific application.

2.2.1 Shack-Hartmann wavefront sensor (SHWFS)

The SHWFS was originally developed in the late 1960s by Dr. Roland Shack and Dr. Ben Platt [84]. The US Air Force posed the problem of improving satellite images, which were taken from the earth, to the Optical Sciences Center (OSC) at the University of Arizona. The Shack-Hartmann or Hartmann-Shack Sensor consists of a two dimensional lenslet array and an optical two dimensional detector (see Fig. 2.4 for the functional principle of the SHWFS). The image plane corresponds to the two dimensional detector, a charge-coupled device (CCD) or complementary metal oxide semiconductor (CMOS) image sensor, depending on the requirements and spectrum/wavelength of the wavefront.

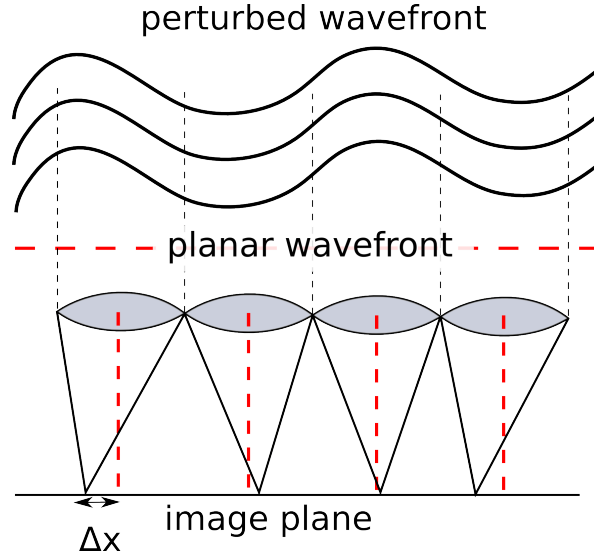


Figure 2.4: Functional principle of the SHWFS in x-direction [67], the red line marks the case when a planar wavefront is sensed with the SHWFS, the black line stands for an exemplarily disturbed wavefront

The basic principle was primarily developed by Johannes Hartmann in the year 1900, called Hartmann-test. The principle is the geometric-optical determination of a local tilt of a wavefront. The main difference is that Shack and Platt replaced the apertures by an array of lenslets. This arrangement has the advantage that the light is focused onto the focal/image plane. The wavefront is separated into a set of discrete

tilts. Measuring the discrete tilts of the wavefront offer measuring the derivative of the wavefront's phase, the aberration. The tilts are the average derivative of the wavefront over the corresponding lenslet; therefore these slopes are sufficient to determine the wavefront without the piston.

The SHWFS is able to measure the optical wavefront aberration. The device consists of an array of micro lenses (lenslets) and a detector, in general, a CCD or CMOS image sensor. All lenslets of the micro lenses array have the same focal length and aperture size. The detector is placed at the focal plane of the lenses. When an ideal planar optical wavefront incident on an optical plano-convex thin lens (along its optical axis), the result is a single intensity point when considering geometrical optics. Any wavefront distortion in the incident wavefront will lead to a displacement of the spots from its original locations. Therefore when regarding the micro lenses array, the average wavefront distortion of the area covered by the micro lenses is measured. Each spot comprises information about the average distortion of the part of the wavefront corresponding to the lens. The average/mean distortion is measured by the location of the spot on the sensor.

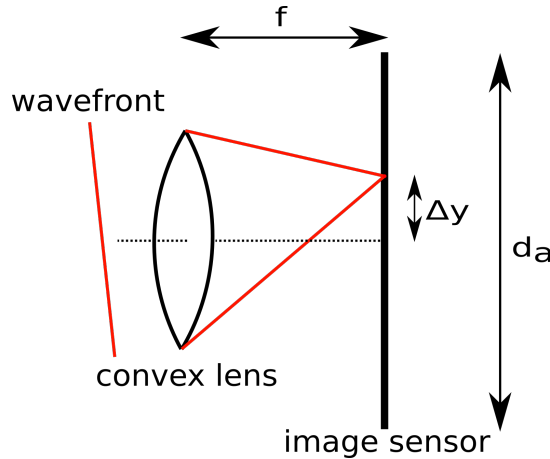


Figure 2.5: Single lenslet of an SHWFS

Fig. 2.5 shows the measurement principle in y-direction for one sub-aperture; the same applies for the x-direction. An enormous advantage and feature of the SHWFS is that the measurement of the wavefront is nearly completely achromatic. The size of the spots on the focal plane for each lens is correlated with the wavelength as the refraction index of the lenses is depended on the wavelength (Sellmeier equation [54]). Therefore the slopes itself are not dependent on the wavelength. Additionally, the SHWFS works on non-point (extended) sources as well. Further, a linear relationship between the mean slope of the wavefront and the displacement of the spot on the image sensor exists. Let $\phi(\vec{r})$ denote the continuous wavefront phase. Then the centroid (center of mass) in x-direction, measured by a single lenslet of the SHWFS,

may be calculated as follows

$$x_{ij} = \frac{1}{A} \int_{ij\text{-th sub-aperture}} \frac{\partial \phi(\vec{r})}{\partial r_x} d\vec{r}, \quad (2.1)$$

where A denotes the area of the sub-aperture. Analogously, the same may be done for obtaining the slope in y-direction. The reference positions $r_{x_{ij}}$ for the ij -th sub-aperture are measured while a flat wavefront is applied. The correlation between slopes and the focal length are as follows

$$\Theta_{x_{ij}} = \frac{\Delta x_{ij}}{f} \quad (2.2)$$

$$\Theta_{y_{ij}} = \frac{\Delta y_{ij}}{f}, \quad (2.3)$$

where Δx_{ij} and Δy_{ij} are the deviations between the reference of the ij -th indexed sub-aperture and the calculated centroids ($\Delta x_{ij} = x_{ij} - r_{x_{ij}}$ and $\Delta y_{ij} = y_{ij} - r_{y_{ij}}$). f denotes the focal length of the lens of each sub-aperture, whereas d_a is the assigned sub-aperture size in y-direction. Usually, the sub-aperture is chosen as a square, such that the dimension is the same for x- and y-direction ($A = d_a^2$ for Fig. 2.5).

The determination of the centroid of each spot for every lenslet, based on the acquired camera sensor data, is investigated in Sec. 3.2 or [39]. For more detailed information about the SHWFS as well as the reconstruction, either Chap. 3 or appropriate literature such as [39, 95, 74, 61] is recommended.

Drawbacks of the SHWFS

Beside the prevalent advantages of an SHWFS, also some limitations arise due to the functional principle of the SHWFS.

The sensitivity of the SHWFS is limited by the ability of the sensor (commonly a CCD-array or CMOS sensor) to determine the exact position of the diffracted spots. On the one hand, the error in determining the diffracted spot location is given by factors such as detector noise, sensitivity of the sensor, photon noise, coherent optical cross talk between diffracted spot location and CCD/CMOS digitization error. On the other hand, the algorithms to determine the centroids of the spots and the separation/ordering process itself [43, 44, 103, 39] affect the accuracy and sensitivity. More detailed information about general limits of wavefront sensing, especially for the SHWFS but also for other WFSs, is given in [38].

2.2.2 Curvature wavefront sensor

The curvature WFS has been developed by François Roddier in 1988. The idea of Roddier was to couple a curvature sensor and a bimorph deformable mirror directly such that no additional calculations is required, i.e. no phase reconstruction based on the slopes as it would be the case when using an SHWFS.

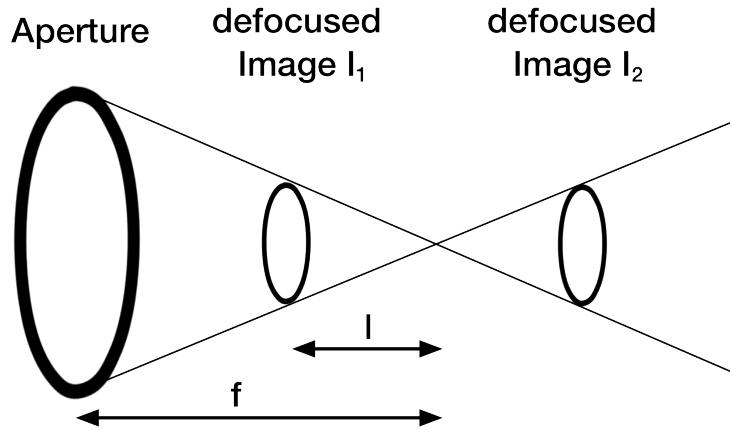


Figure 2.6: Functional principle of a curvature WFS

The curvature of a wavefront, or its phase ϕ , is calculated by applying the Laplacian operator, $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ to the wavefront. The curvature WFS is achromatic and the sensitivity is inversely proportional to the de-focusing l^3 . Larger de-focusing is required to measure the wavefront with a higher resolution, whereas the sensitivity of the sensor will be reduced accordingly. Therefore, the curvature WFS has problems when high-order aberrations are required to be sensed.

2.2.3 Pyramid wavefront sensor

The pyramid wavefront sensor (P-WFS) has been developed by Italian astronomers [88]. Instead of an array of lenslet, a transparent pyramid is placed in the focal plane of the center (see Fig. 2.7). The pyramid splits the incoming beam star-shaped into four parts onto the detector plane. Each beam gets deflected and these beams, in turn, form four images of the wavefront on a typically CCD/CMOS detector which is the same for all four images. The optical setup of the P-WFS is similar to the Foucault knife-edge test.

It is noted that a tunable modulation is necessary to operate the WFS in the linear regime and to sense aberrations accurately [88]. Nevertheless, there have been approaches presented which do not require modulation at all, but face other difficulties [89].

The functional description of the P-WFS (see Fig. 2.7) is as follows: Supposing that the light source is extended and geometrical optics is applied. Then, a wavefront slope at some sub-aperture changes the source position on the pyramid. Therefore, the light flux changes which is detected by the four pixels which would otherwise be equal, regarding the intensity. By calculation of the normalized intensity differences, two signals proportional to the wave-front slopes in two directions are determined. The sensitivity of a P-WFS depends on the dimension of the source. The P-WFS can be regarded as an array of quad-cells, and thus is similar to an SHWFS. There exists

³ www.ctio.noao.edu/~atokovin/tutorial/part3/wfs.html

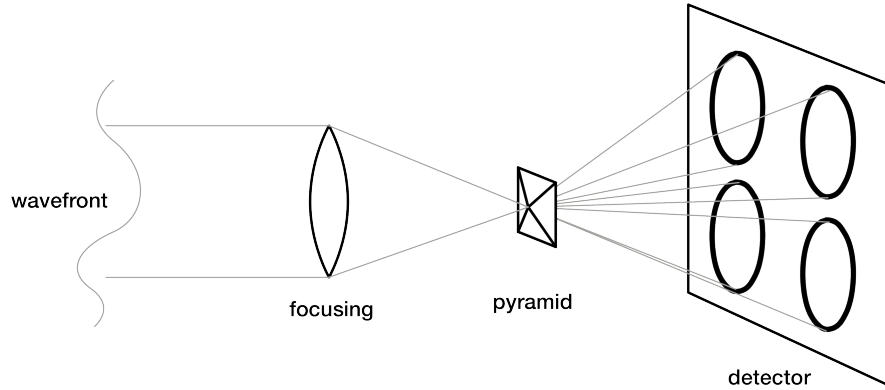


Figure 2.7: Exemplary functional principle of the P-WFS

some duality with the SHWFS; the pixels of an SHWFS correspond to the lenslets in a P-WFS. Therefore, each sub-aperture is detected by four CCD/CMOS pixels.

Considering a point source with diffraction effect, the intensity distributions in the four pupil images become a non-linear function of the wavefront. This means the P-WFS does not measure slopes any more. In order to achieve the linearity again, the point source is rapidly moved over the pyramid edge typically in a circular pattern. The effect is a smearing of the point source because the signal is integrated over one or more wobble cycles.

One advantage of the P-WFS is that no lenslet array is necessary. Thus, the sub-apertures are defined by the detector pixels. Further, a tradeoff between sensitivity (smaller modulation) and linearity (larger modulation) can easily be realized. Even at smaller amplitudes, the sensitivity of a P-WFS can be higher than that of an SHWFS [98]. Desiring high sensor rates, the modulation of the pyramid must be adjusted accordingly to guarantee a linear behavior, which is necessary when using the wavefront information for controlling e.g. the wavefront error. It is even possible that multiple pyramids are placed in the focal plane, when e.g. multiple faint guide stars have to be analyzed with a single WFS. Nevertheless, till now, this application has not been realized and is likely only of interest for astrophysical researchers.

2.2.4 Shearing interferometer

The shearing interferometer is an extremely simple setup to observe interference. The fundamental idea of an shearing interferometer is that a duplicate of the beam under test is generated and interfered with the original beam. Between the two different beams, some shearing (see Fig. 2.8) is induced. Commonly, the shear is created by using an uncoated plane parallel plate of transparent glass having a width of ρ .

Usually, at the beam incident of about 45° ; part of the beam is reflected at the front and also at the back surface. Through the small difference in the path (small lateral shift ρ), both beams interfere with each other. Having small shifts, the phase difference is proportional to the first derivative, called slope of the wavefront, as long

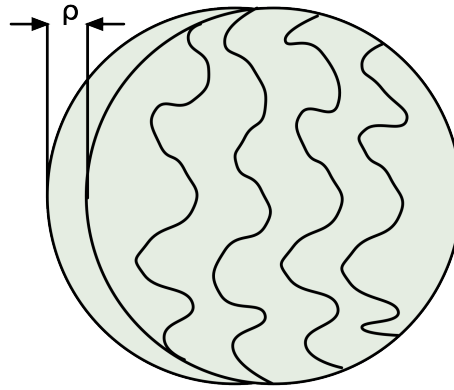


Figure 2.8: Shearing interferometer with small lateral shift ρ

as the shear distance is small compared with the scale of the disturbance. Therefore, the measurement signal is comparable with that of an SHWFS. To measure both x-/y-slopes, two individual shears in the orthogonal directions are required. The phase differences are converted through the interference into intensity variations.

The first successful AO system named Real-Time Atmospheric Compensator (RTAC) [39] build in 1973, was based on a shearing interferometer WFS. Nevertheless, this approach has been now completely abandoned in favor of other wavefront sensing techniques, as other types of interferometers were proposed for wavefront sensing. Some of them have the property that their signals are directly proportional to the phase. This means that no reconstruction of the phase, and therefore no calculation for obtaining the wavefront, is required. Nonetheless, these methods are limited in their dynamical range.

2.2.5 Holographic wavefront Sensor

The holographic wavefront sensor (HWFS) is a modal wavefront sensing device instead of the SHWFS which is a zonal WFS. A hologram is recorded between an object beam having e.g. a minimum specific aberration (see Fig. 2.9) and a reference beam focused at a point 'A' where a detector is located. Afterwards, a second hologram is recored on the the same medium, typically a holographic optical element (HOE), using an object beam with the maximum amplitude of the same specific aberration and a reference beam focused at a point 'B'. The defined aberrations should be e.g. Zernike polynomials, or in general, orthonormal polynomials. For more detailed information about Zernike polynomials and their characteristics, see Sec. 3.1. The use of these aberrations allows to describe a wavefront quite efficiently without infinite holograms, which would be required for an exact description. Nevertheless, recently an HWFS has been presented which is able to determine the acting values for a deformable mirror without any need of a computer [5].

If multiple holograms are recorded in the HOE, e.g. the first Zernike polynomial with a maximum and minimum amplitude being each focused at a different point,

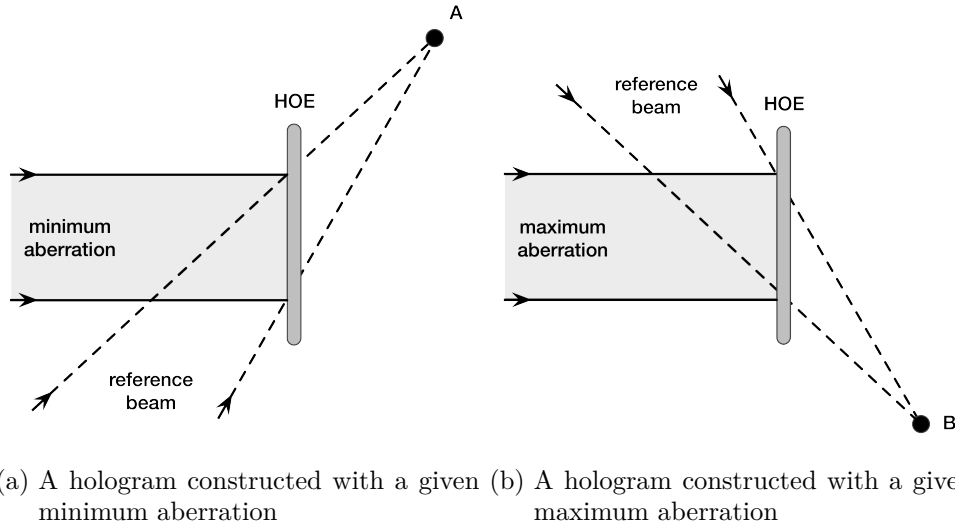


Figure 2.9: Multiplexing two holograms to detect the given specific aberration

then an object beam (generic aberration) (see Fig. 2.10), illuminates point 'A' and 'B'. Based on the intensity of each point, the occurrence of the specific aberration used for recording the hologram can be determined. Here, by evaluating the intensity information at 'A' and 'B', the amount in terms of the aberration, used for recording the hologram, can be calculated as follows

$$p_i = \frac{I_A - I_B}{I_A + I_B}$$

whereas p_i is the calculated coefficient of the specific recorded aberration.

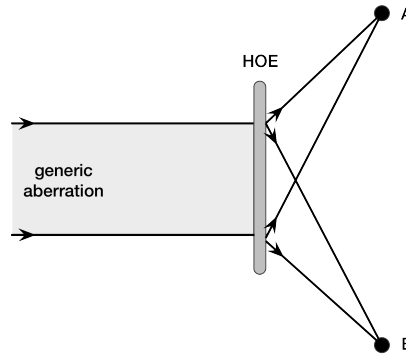


Figure 2.10: Complete wavefront characterization when two holograms are being multiplexed

For more detailed information about the HWFS, see [4, 30, 123, 5].

2.3 Experimental Optical Setup

The experimental optical setup has been designed to verify the control strategies by application of predefined repeatable disturbances. The setup has been constructed as a breadboard design (see Fig. 2.11); to introduce specific disturbances the deformable mirror (DM1) is used. The tip/tilt mirror is from Physik Instrumente (S-330.2SL), whereas the DM2 as well as the DM1 are deformable mirrors designed and manufactured by the Fraunhofer IOF Jena. To characterize the wavefront, two WFS are available; on the one hand, the SHWFS HASOTM3 Fast from Image Optics having a 14×14 micro lens array with a maximum frame rate of 905 Hz (WFS1) and on the other hand, the SHSCam HR-130 from Optocraft GmbH having a 68×51 micro lens array with a frame rate of 10 Hz (WFS2). For more information about the WFS1, see Appendix A.9. Due to the construction, it is possible to generate specific distur-

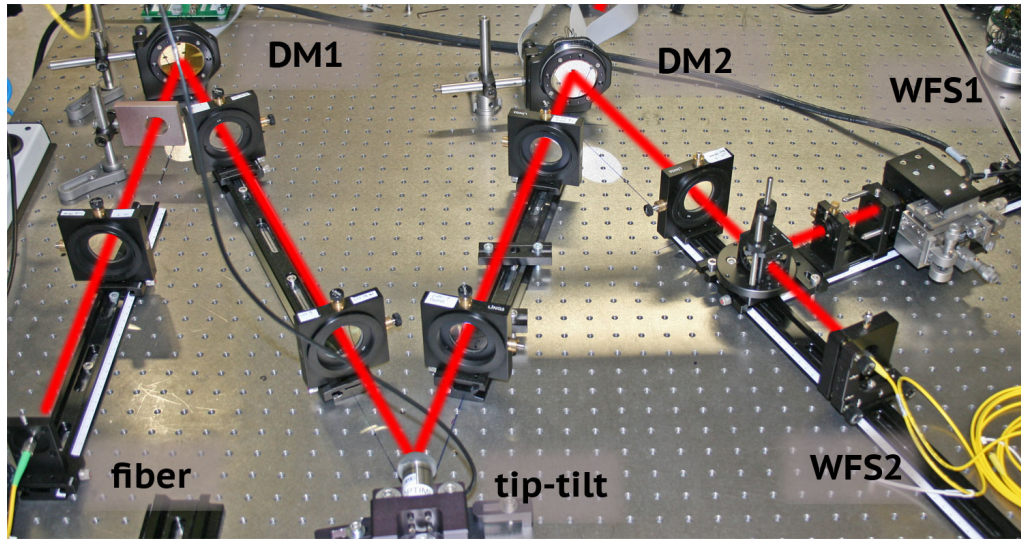


Figure 2.11: Breadboard platform of optical setup for specific disturbance injection and controlling of these disturbances with a deformable mirror

bances while eliminating those with the DM2. Of course, the controller calculating the acting values for the DM2, is supposed to know nothing about the characteristics of the DM used for the disturbance (DM1).

As laser, the HRP170 from Thorlabs GmbH is applied. The laser beam is feed in a fibre glass and collimated afterwards. In the following, the beam diameter is adjusted by a succeeding diaphragm. One advantage of the optical setup is that each mirror plane is conjugated to both measurement planes. Using the 1 : 1 telescope, both apertures are identical regarding their diameter and the beam ratio remains identical. The 1 : 14 telescope is required to match the aperture of the DM (25 mm) to the 1.6×1.6 mm aperture of the Image Optics HASOTM3 Fast (WFS1).

The cube beamsplitter (CBS) is a 50 : 50 beam splitter to allow the use of both WFS in parallel. During the design of the optical setup, special care has been given to minimize the incidental angle to the optical elements. The actuator layout of the

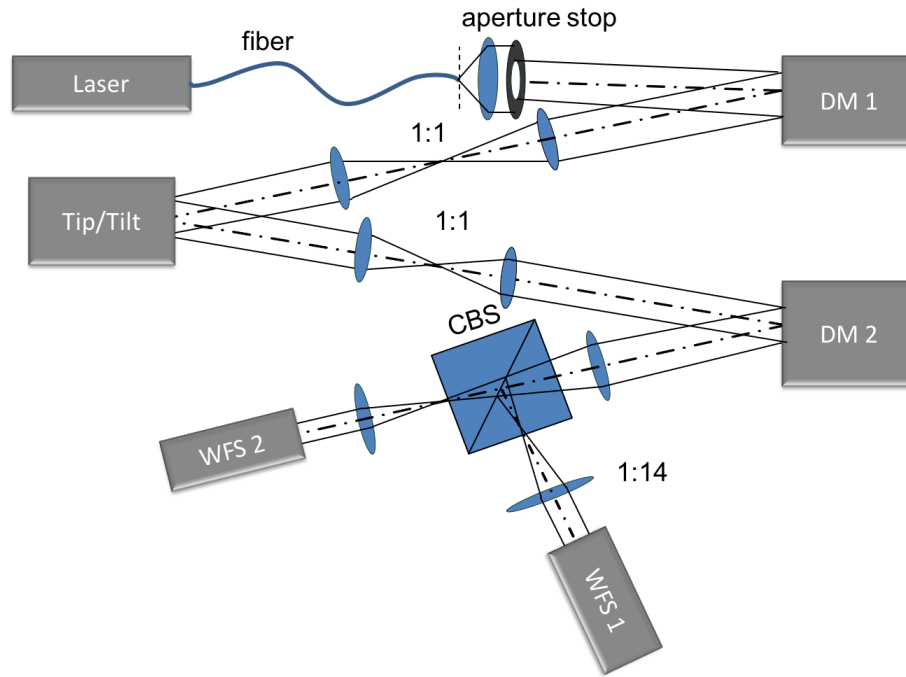


Figure 2.12: Geometric schematic of optical setup with all telescopes employed

DMs has been optimized for a diameter of 20 mm respectively 22.5 mm; therefore a smaller diameter would result in worse performance regarding the compensation of disturbances of the wavefront.

3 Real-Time Wavefront Measurement Algorithm

I think there is a world market for maybe five computers.

THOMAS WATSON, CHAIRMAN OF IBM, (1943)

This chapter addresses the algorithms for evaluating a Shack-Hartmann wavefront sensor (SHWFS) in real-time. For correct modeling of an SHWFS and to be able to incorporate the specific behaviors regarding delay/latency¹, as well as the limitation of wavefront measuring due to the measuring principle of an SHWFS, a detailed analysis of the complete SHWFS is necessary and useful.

In general, commercially available wavefront sensors (WFSs) usually are not designed for high-speed real-time wavefront measurement due to their hardware and/or related software. E.g. the software of the available HASOTM3 Fast² SHWFS is only available for Windows operating systems (OSs) which is neither real-time capable nor can guarantee any latency or computational power for specific applications. Some SHWFS manufacturers provide an implementation routine for the evaluation with the PCI extensions for instrumentation (PXI) laboratory virtual instrumentation engineering workbench (LabVIEW) platform from National Instruments (NI), but almost no manufacturer publishes detailed information concerning the applied algorithm for evaluation of the SHWFS. Of course, the algorithms behind the evaluation of the SHWFS are intellectual property (IP) of each company, however, for identifying the limitations of the underlying algorithms it is essential to know the functional principle of the applied algorithm. Hence, the received data of the SHWFS has often to be regarded with doubt as the maximum specifications are exceeded or even not defined for specific cases. During conceptual design of high-speed control systems, the most critical part regarding the SHWFS is the latency itself followed by the uncertain processing time due to an e.g. non real-time capable OS. The result is showing an uncertain latency with some overall minimum latency. The latency has a particular significance regarding the design of a control-loop because it has a strong impact on the performance of the closed-loop system when using the WFS as the sensor for measuring the wavefront, see Chap. 4.

In light of the mentioned reasons, a lot of work has been put into analyzing and

¹ The term latency is a synonym of delay, therefore being equivalent; till the end of the document, only the term latency is further used.

² http://www.imagine-optic.com/iop_products_wavefront-analysis_HASO-Fast_shack-hartmann-wavefront-sensor_en.php

developing new algorithms, based on different published algorithms and concepts for evaluating the spot pattern of an SHWFS as well as the spot detection itself. Bearing in mind the predefined goal of employing a rapid control prototyping (RCP) system it is necessary to evaluate different control concepts in an acceptable time. High performance has to be guaranteed as well, therefore, the application of an field programmable gate array (FPGA) for evaluating the SHWFS is self-evident. The benefit of an FPGA solution is that the determination of the individual centroids likewise as for spot pattern ordering can be accelerated by a parallel approach. The algorithms for evaluating the wavefront do not have to be adjusted when changing e.g. the controller, hence they are highly suitable to be implemented in hardware. On account of this, novel algorithms such as the application of connected component labeling (CCL) for determining the centroids and advanced spot ordering methods such as the spiral algorithm have been analyzed and implemented.

First, Sec. 3.2 deals with the conventional approach for evaluating an SHWFS and discusses the drawbacks of the method briefly. Numerous additional methods such as phase imaging or curvature sensing [61] may also be used to measure the wavefront, but either they are very expensive or do not have the desired performance regarding speed and latency. A general overview of the wavefront measuring techniques is given in Sec. 2.2. In Sec. 3.1 a rough overview concerning wavefront reconstruction, based on the measured slopes of the SHWFS, is given.

Afterwards, Sec. 3.3 presents in detail newly developed algorithms for evaluating an SHWFS which are suitable to be implemented efficiently on an FPGA. The focus of the algorithms is clearly put on high-throughput, thus achieving high sensor data rates combined with a low-latency. For determining the centroids without saving the image from the SHWFS, the application of single-pass CCL is presented, combined with a simple spot pattern ordering. Additionally, a slightly modified spiral ordering algorithm is presented and analyzed which also has a deterministic run-time and allows to segment the centroids correctly when compared to the original spiral algorithm. Simulations and experimental data of the algorithms are compared with previously published algorithms. Each of the developed algorithms has been implemented on a low-cost/mid-range FPGA, based on VHSIC hardware description language (VHDL).

3.1 Wavefront Reconstruction*

Based on the local wavefront slopes, $\Theta_{x_{ij}}$ and $\Theta_{y_{ij}}$ (see Equ. (2.2) and (2.3)), the wavefront can be reconstructed by integration of the gradient measurements. As a result, the reconstructed phase of the wavefront, assuming monochromatic light emitted from a point source in an infinite distance, ideally has the same value for the overall wavefront. There are reported two distinctive wavefront reconstruction methods in the literature, the zonal and modal approach [82, 74, 39].

The zonal wavefront reconstruction determines the phase at specific points e.g. by numerical integration or calculation of a least squares solution for a linear system of equations. In contrast, the modal wavefront reconstruction uses a set of orthogonal polynomials such as Zernike polynomials. The measured wavefront data is fitted to this set of polynomials until a desired order of the polynomial set. Due to the orthogonality of the polynomials, this may be realized in parallel. Although the zonal approach was very popular in the early beginning of adaptive optics (AO) systems, nowadays the modal approaches are known to be better in terms of the error propagation. But still, they need more computational power in general. Based on orthogonal polynomials the wavefront is described completely, not only at specific points, as during the zonal reconstruction. Additionally, specific properties such as tip/tilt may be identified directly when Zernike polynomials are used.

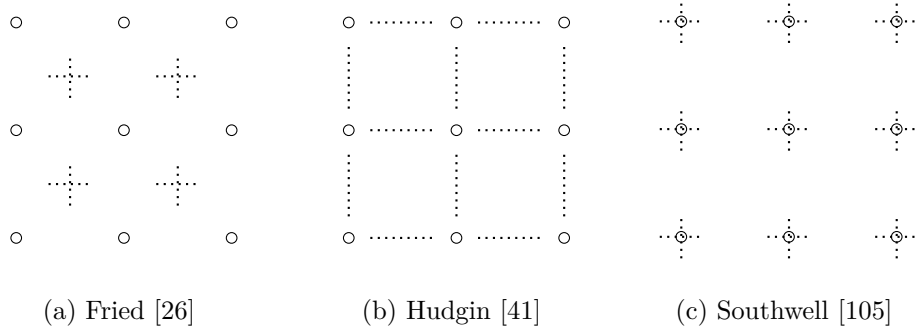
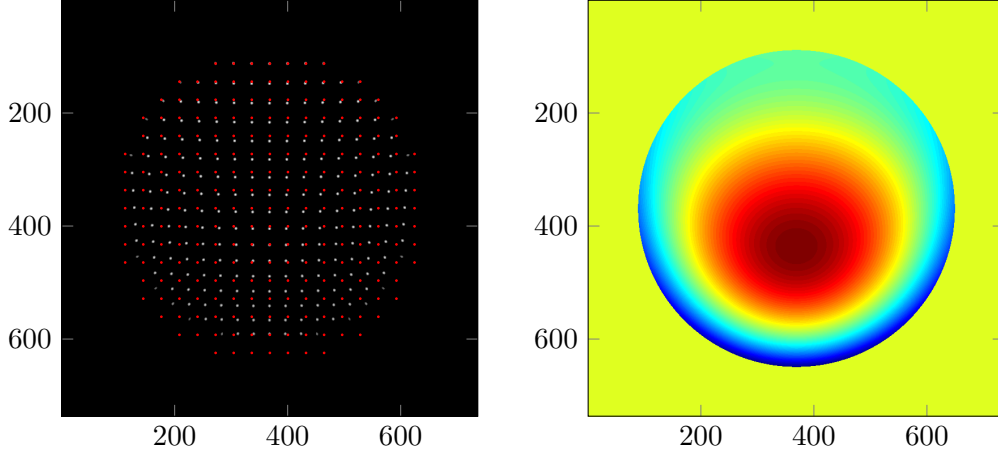


Figure 3.1: Different reconstruction configurations; the horizontal dots represent the positions of the x-slope and the vertical dots correspond to the y-slope positions. The curls are the points where the phase is estimated.

In total, there are three main geometries, called Fried [26], Hudgin [41] and Southwell [105], see Fig. 3.1 for a visualization. Based on these geometries, the wavefront slope can be approximated by finite differences but the geometries are only an approximate model for the SHWFS case. In the past, Fried and Southwell have been used for the SHWFS case successfully, either applied for the modal or zonal approach.

In Fig. 3.2a, exemplarily sensor data of the SHWFS (colored in white) and corresponding reference points (colored in red) are shown. Based on these two sources of information, the slopes can be determined. Fig. 3.2b shows the reconstructed wavefront using modal reconstruction based on Zernike polynomials exemplarily.



(a) Exemplarily camera image of the SHWFS

(b) Reconstructed wavefront

Figure 3.2: SHWFS camera image colored in white with reference spots in red (a). The reconstructed wavefront is based on the determined slopes (b).

Zonal reconstruction

The simplest zonal wavefront reconstruction method is called linear integration. Starting at one edge of the wavefront slope data and defining the wavefront height value to zero initially, the height of the wavefront in the next adjacent location is calculated as the previous wavefront height added to the product of the previous slope multiplied with the distance between the two sub-apertures. The height of the wavefront corresponds to the phase of the wavefront.

For the x-direction, the equation is as follows

$$\Phi_{n,m}^x = \Phi_{n-1,m}^x + \frac{\partial \Phi_{n-1,m}}{\partial x} d_a \quad (3.1)$$

where $\Phi_{n,m}$ is the wavefront surface height at position (n,m) and d_a denotes the distance between the sub-apertures. For the y-direction, x has only to be substituted by y . When the linear integration in both directions has been carried out, the overall wavefront height is simply

$$\Phi_{n,m} = \Phi_{n,m}^x + \Phi_{n,m}^y. \quad (3.2)$$

The basic problem of the simple linear integration method are measurement errors. Almost at any time the slope measurements are noisy and due to the integration the resulting errors propagate through the calculation. Therefore the resulting wavefront height is likely to be noisy as well, but the reconstruction needs little time due to the low complexity. Supplementary, the reconstructed phase is consistent in most of the cases. However, the error between the reconstructed wavefront and the measured slopes is neither minimized nor forced to be somehow bounded. Thus, nearly arbitrary deviations may occur between the reconstructed and real phase of the wavefront. As it has been discussed when presenting the different geometries for obtaining the

slopes via finite differences, these models are only approximations. Thus the Hudgin geometry (visualized in Fig. 3.1b) is introducing errors in the reconstructed wavefront too when being applied for the SHWFS case.

Another popular zonal method is using the Southwell configuration [105], which also results in very fast computation and at the same time providing high accuracy. The main idea of the Southwell reconstructor is to minimize the error between the reconstructed wavefront and the measured wavefront slopes. To accomplish the desired goal, the wavefront slopes are scanned through the entire grid and in the same time, the wavefront height is calculated at each point based on the average height predicted by each neighbor as follows

$$\Phi_{i,j} = \sum_{n=-1}^1 \sum_{m=-1}^1 I_{i+n,j+m} \left(\Phi_{i+n,j+m} + \left(\frac{\frac{\partial \Phi}{\partial x_{n,m}} + \frac{\partial \Phi}{\partial x_{i+n,j+m}}}{2} \right) d_a \right), \quad (3.3)$$

where $\Phi_{i,j}$ is the wavefront at lenslet (i,j) , d_a is the width of the lenslets and $I_{i+n,j+m}$ the intensity of the spot at the corresponding lenslet. Due to the intensity weighting, which can be toggled on and off by the user, a measure for the validity of a wavefront slope is available. In general, a spot corresponding to a lenslet with high intensity has a more accurate slope measurement than a spot with low sensitivity. The Southwell algorithm may take many iterations through the measurements of the SHWFS data to achieve an accurate result. Because of the structure, the algorithm will always converge but the absolute maximum number of iterations is equal to the number of applied lenslets. In reality the convergence will be much faster in almost any case.

Based on the configuration shown in Fig. 3.1c, the reconstruction may also be formulated as a least-squares problem. This formulation has been used for reconstructing the wavefront in the WFS plugin in QTScope³. A MATrix LABoratory (MATLAB) implementation of this method is presented in [74]. The average slope measurement of the SHWFS is given by

$$\frac{\Theta_{i,j+1}^x + \Theta_{i,j}^x}{2} = \frac{\phi_{i,j+1} - \phi_{i,j}}{d_a} \quad (i = 1, \dots, N; \quad j = 1, \dots, N-1) \quad (3.4)$$

$$\frac{\Theta_{i+1,j}^y + \Theta_{i,j}^y}{2} = \frac{\phi_{i+1,j} - \phi_{i,j}}{d_a} \quad (i = 1, \dots, N-1; \quad j = 1, \dots, N), \quad (3.5)$$

where N represents the number of lenslets in x- and y-direction. Equ. (3.4) and (3.5) can be solved by applying standard least-squares methods [41], solving

$$C\Theta = E\Phi, \quad (3.6)$$

where Θ is a column vector of the slope measurements, Φ a column vector of the unknown wavefront and C , E are sparse rectangular matrices. For a 2×2 sensor,

³ <https://github.com/steffenmauch/QTScope>

Equ. (3.6) may be written as follows [74]:

$$\begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} S_{1,1}^x \\ S_{1,2}^x \\ S_{2,1}^x \\ S_{2,2}^x \\ S_{1,1}^y \\ S_{1,2}^y \\ S_{2,1}^y \\ S_{2,2}^y \end{bmatrix} = \frac{1}{d_a} \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} W_{1,1} \\ W_{1,2} \\ W_{2,1} \\ W_{2,2} \end{bmatrix} \quad (3.7)$$

If matrix E is square and full rank, as in Equ. (3.6), the solution may be obtained by multiplying with the inverse of E , thus $\Phi = E^{-1}C\Theta$. Often, however, the number of measurement points is larger than the number of estimated points of the wavefront which results in an overdetermined system. Therefore matrix E is no longer a square matrix but instead a rectangular matrix with dimension $M \times N_w$, $M > N_w$, where $M = 2N(N - 1)$ (N are the number of lenslets in x- and y-direction) and $N_w = N^2$ is the number of wavefront values. Assuming that matrix E has full rank, using the pseudo-inverse, the solution reads

$$\Phi = (E^T E)^{-1} E^T C \Theta. \quad (3.8)$$

Whenever E is an $M \times N$ matrix with rank deficiency, a solution of Equ. (3.6) with minimum norm is desired. The wavefront piston can never be recovered based on wavefront slopes as the slopes are the derivatives of the wavefront. To obtain an optimal solution, typically the singular value decomposition is applied. Matrix E is decomposed in

$$E = U D V^T \quad (3.9)$$

where U and V are orthogonal matrixes and D is a diagonal matrix having the singular values of E . Then, the solution is

$$\Phi = V D^{-1} U^T C \Theta. \quad (3.10)$$

Modal reconstruction

In contrast to the zonal reconstruction, modal reconstruction allows to have a continuous representation of the wavefront; not only at specific points predefined by the wavefront grid. In addition, by choosing an appropriate set of orthonormal polynomials (e.g. Zernike polynomials), dominating aberrations may easily be determined by inspecting the coefficients of the individual polynomials. As standard set when analyzing optical elements, originally Zernike polynomials have been used. The Zernike

polynomials are a product of a radial and azimuthal part. Therefore, they are written in the form $Z_n^m(r, \theta)$. The radial orders are positive integer values $n = 0, 1, 2, \dots$ whereas the azimuthal indices range from $m = -n$ to $m = +n$, and $m - n$ is even. In Tab. 3.1, the first six normalized Zernike polynomials in polar coordinates are listed and Fig. 3.3 visualizes each polynomials individually.

Zernike polynomials are a set of polynomials which are used for describing a surface. One of its major properties is that they are orthogonal over the unit circle. This means that, over the unit circle, the inner product of any two Zernike polynomials is zero. For example, inspecting Fig. 3.3 for tip-/tilt it is obvious that the inner product over the unit circle is zero. Different conventions regarding the numbering scheme, normalization etc. exist and are well described in [101].

Furthermore, there exists publications which use the Fourier series for modal wavefront reconstruction, in view of the highly efficient fast-fourier transformation (FFT) algorithm. Nevertheless, the Fourier series is only orthonormal over rectangular domains and not on a circular one. For this reason, methods have been proposed to prevent problems by using the iterative Fourier reconstruction [74, 96, 22, 86]. Other

i	n	m	$Z_n^m(r, \theta)/Z_i(r, \theta)$	name
0	0	0	1	piston
1	1	-1	$2(\frac{r}{R}) \cos(\theta)$	tip
2	1	1	$2(\frac{r}{R}) \sin(\theta)$	tilt
3	2	-2	$\sqrt{3} [2(\frac{r}{R})^2 - 1]$	focus
4	2	0	$\sqrt{6} (\frac{r}{R})^2 \sin(2\theta)$	astigmatism
5	2	2	$\sqrt{6} (\frac{r}{R})^2 \cos(2\theta)$	astigmatism

Table 3.1: First six low order normalized Zernike polynomials in polar coordinates

sets of orthonormal polynomials such as Legendre, Tschebyschow, Jacobi or Hahn polynomials may also be used, but they are not common in this context. Orthonormal polynomials show the following five properties:

- The wavefront mean value is represented by the piston coefficient.
- The Zernike coefficients are independent of the number of terms used for the wavefront reconstruction.
- Each Zernike polynomial has a mean value of zero.
- Each Zernike polynomial has a minimum variance. Thus, the Zernike polynomials representing the wavefront have also a minimum variance.
- Excluding the piston coefficient, the root-mean square (RMS) error of the wavefront is the square root of the sum of the squares of Zernike coefficients (depends on whether the normalized Zernike polynomials are used or not).

Derivatives of Zernike polynomials are not orthogonal. Therefore, in general the derivatives cannot be fitted independently. However, there are several publications

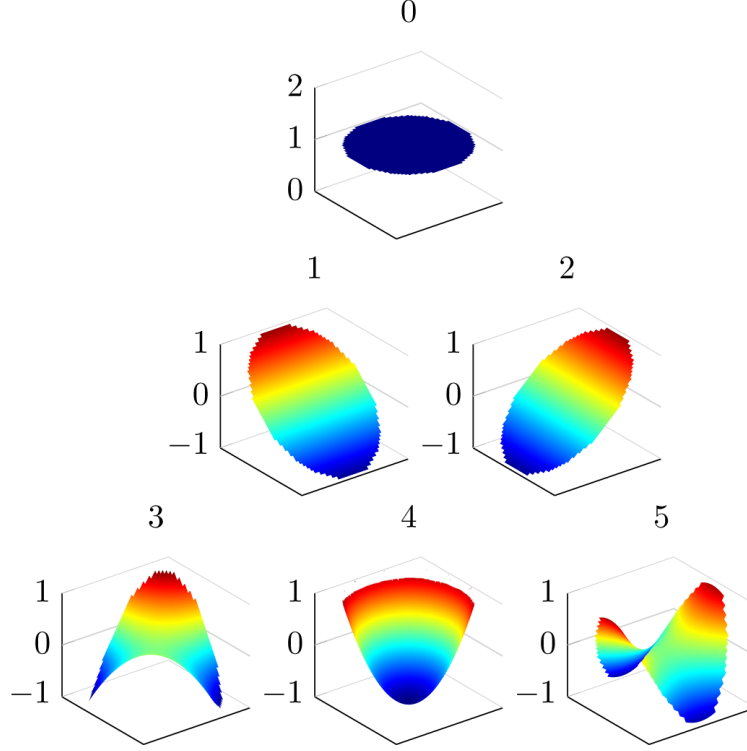


Figure 3.3: Illustration of the first six low order Zernike modes

(e.g. [1, 29]) which introduce sets of orthogonal polynomials having the property that the inner product depicts the Zernike coefficients directly.

Using the slope measurements from the SHWFS, the modal wavefront reconstruction fits these measurements to coefficients of derivatives with respect to a given set of basis functions. Sampled Zernike polynomials do not form an orthogonal basis anymore. Therefore, [75] proposes a method based on the QR decomposition to orthogonalize the set of derivatives. Loosing orthogonality has as consequence, in general, a bad condition number of matrix E , see Equ. 3.15. The consequence is an amplification of the noise when the following steps are traversed.

Therefore, the maximum polynomial order of the used Zernike have to be chosen according to the available slopes. Otherwise, the condition number of E will be huge and problems known from one dimensional high-order polynomial fits will likely arise.

Up to now, only the advantages and disadvantages of using Zernike describing the wavefront are discussed. In the following, the mathematical procedure is shown: The approach to describe a wavefront Φ based on the Zernike is as follows. Take

$$\Phi(R\rho, \theta) = \sum_{i=1}^J a_i F_i(\rho, \theta), \quad (3.11)$$

where the piston is neglected, thus, begin with $i = 1$. J is the maximum polynomial order of the corresponding set. The use of a polar coordinate system is beneficial because the wavefront is determined over the unit circle. Calculating the derivatives

with respect to x and y of Equ. (3.11) for the l -th of in total k sampling points. Then, the following equations are obtained

$$\left. \frac{\partial \Phi(R\rho, \theta)}{\partial x} \right|_l = \sum_{i=1}^J a_i \left. \frac{F_i(\rho, \theta)}{\partial x} \right|_l \quad (l = 1, \dots, k) \quad (3.12)$$

$$\left. \frac{\partial \Phi(R\rho, \theta)}{\partial y} \right|_l = \sum_{i=1}^J a_i \left. \frac{F_i(\rho, \theta)}{\partial y} \right|_l \quad (l = 1, \dots, k). \quad (3.13)$$

Rewriting both equations in a matrix representation, the result is

$$\Theta = Ea \quad (3.14)$$

where Θ is a column vector of the slope measurements, a is a column vector of the unknown coefficients and E is

$$E = \begin{bmatrix} \left. \frac{F_1(\rho, \theta)}{\partial x} \right|_1 & \left. \frac{F_2(\rho, \theta)}{\partial x} \right|_1 & \cdots & \left. \frac{F_J(\rho, \theta)}{\partial x} \right|_1 \\ \vdots & \vdots & \ddots & \vdots \\ \left. \frac{F_1(\rho, \theta)}{\partial x} \right|_k & \left. \frac{F_2(\rho, \theta)}{\partial x} \right|_k & \cdots & \left. \frac{F_J(\rho, \theta)}{\partial x} \right|_k \\ \left. \frac{F_1(\rho, \theta)}{\partial y} \right|_1 & \left. \frac{F_2(\rho, \theta)}{\partial y} \right|_1 & \cdots & \left. \frac{F_J(\rho, \theta)}{\partial y} \right|_1 \\ \vdots & \vdots & \ddots & \vdots \\ \left. \frac{F_1(\rho, \theta)}{\partial y} \right|_k & \left. \frac{F_2(\rho, \theta)}{\partial y} \right|_k & \cdots & \left. \frac{F_J(\rho, \theta)}{\partial y} \right|_k \end{bmatrix}. \quad (3.15)$$

$\left. \frac{F_i(\rho, \theta)}{\partial y} \right|_l$ denotes the average partial derivative of the basis function $F_i(\rho, \theta)$ at the l -th sub-aperture. The solution of the matrix equation is similar to the least-square solution in Sec. 3.1. The singular value decomposition is used to decompose the matrix E such that $E = UDV^T$ where U, V are orthogonal matrices and D is a diagonal matrix containing the singular values of E .

3.2 Conventional Approach

The conventional approach (also named first moment calculation) assumes that each sub-aperture has its own area on the image sensor. The image sensor is typically a charge-coupled device (CCD) up to now, which gradually is being replaced with complimentary metal oxide semiconductor (CMOS) sensors. Each lens focuses the local wavefront on the image sensor plane. Thus, the centroids of the spots in each area may be calculated by application of the following formulas

$$x_c = \frac{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} x_i I(x_i, y_j)}{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} I(x_i, y_j)} \quad (3.16)$$

$$y_c = \frac{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} y_j I(x_i, y_j)}{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} I(x_i, y_j)}. \quad (3.17)$$

The centroid position in the detection area (for each sub-aperture) are named (x_c, y_c) . M and N is the pixel number in x - and y -direction of the associated sub-aperture, respectively. $I(x_i, y_j)$ is the intensity at the given coordinate (x_i, y_j) . Equ. (3.16) and (3.17) is the optimal estimator for the case when the spots are Gaussian distributed and the noise is Poisson. Using a spatial light modulator (SLM) instead of lenslets [117], the assumption that the distribution is gaussian is not valid anymore. The measurement error of an SHWFS is treated in [19, 103]. In this dissertation, see Sec. 2.2.1 for a brief discussion.

Before calculating the centroid position, thresholding is applied for the image data. If $I(x_i, y_j)$ is smaller than the given threshold value then $I(x_i, y_j)$ is set to zero. This suppresses effects of sensor noise in the conventional approach. Choosing an adequate threshold, the calculated spot position is valid if the assumption of a Gaussian distribution applies.

Formulas Equ. (3.16) and (3.17), are only applicable when known in advance that inside the examined area stays either exactly no or only one spot (see Fig. 3.4a), otherwise the calculation will be invalid (see Fig. 3.4b). The condition that the spot lies inside d_a is satisfied when the maximum tip/tilt of the wavefront is limited by

$$x_{max} = \frac{0.5 d_a}{f} \quad (3.18)$$

$$y_{max} = \frac{0.5 d_a}{f}, \quad (3.19)$$

see also Fig. 2.5.

Therefore, given by Equ. (3.18) and (3.19), the dynamic range of the SHWFS is limited by the focal length f and the length d_a of the area on the image sensor. If a higher dynamic range is needed then the focal length may be adjusted, however, involving simultaneously that the minimum resolution is decreased. Another self-evident approach is that the spot is allowed to leave the associated sub-aperture, but then the conventional approach is not suitable any more.

Claiming that the spots stay inside the associated sub-apertures can be very restrictive in an experimental setup. The reason is that the defocus of the wavefront is often very strong when not aligned carefully. A huge defocus results in an increase/decrease of the distances between the individual spots and therefore less spots are on the image sensor or the overall distance between the spots is shrinking, thus, the problem of sub-aperture crossing may appear, see Fig. 3.4. Apparently, other aberrations may also lead to spots which do not remain in their corresponding sub-aperture, but defocus is in most cases dominant. Nevertheless the origin of the problem, namely the strict assignment to an area for each spot, remains the same in all cases.

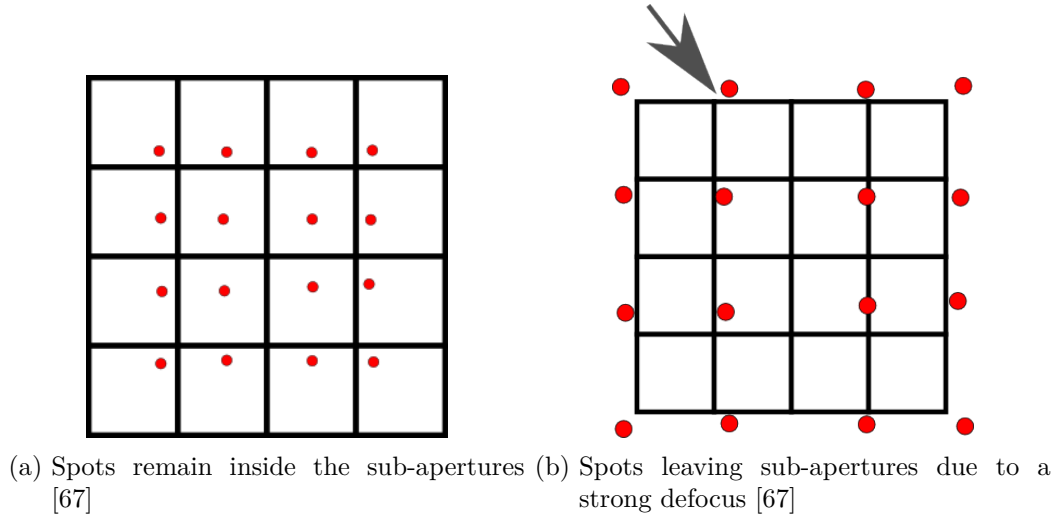


Figure 3.4: Consequence of a huge positive and negative defocus when having a strict area where the spots must remain

3.3 Contributed Algorithms for Evaluation of SHWFS

3.3.1 Overview over SHWFS evaluation approaches

In view of the restrictions of the conventional approach, discussed in Sec. 3.2, the question arises whether there are better algorithms which are not based on specific regions for each spot on the image sensor. This way, the dynamic range may be increased compared to the conventional approach while having the same sensitivity. Of course, an algorithm with a smaller complexity in comparison with the conventional approach (complexity $O(n)$) is not realizable because the image stream is only available sequentially, thus as a data stream. Therefore, the resulting processing time of the conventional approach is the lowest possible when streaming the image sensor data.

In state-of-the-art approaches either the latency is minimized, disregarding the achievable dynamic range, or the dynamic range is extended, deferring latency issues as well as a possible high-speed implementation. For instance in [47], an FPGA solution is implemented under the assumption that spots cannot leave the associated sub-apertures.

In [116] an improved version of the conventional algorithm is presented which tracks the spots such that a spot leaving his corresponding area on the image sensor is admissible. However, the approach is only applicable if the spots are determined correctly at least once. For initial position recognition there is no solution yet. but when the wavefront is known during start, e.g. having a perfect initial calibration, the problem does not arise. Another drawback is the fact that the spots are not allowed to move faster than the specified frame rate, otherwise the adaptation will likely fail when having a huge disturbance. This happens due to the fact that the algorithm cannot track the movement of the spots anymore. Therefore the algorithm will assign the spots erroneously. In a practical setup, such a restriction may lead to complications where, in general, it cannot be guaranteed that the disturbance is slower than a specified rate.

In [56] prior information of the specific application is used and thus, the spots are not constrained to stay in the view-field of their lenslets. The method presented in [53] is based on a similar method, needing reference centroids for initialization. Additionally, the paper discusses general problems like spot crossing and row crossing that may appear when the spot does no longer remain in the confined area of the single lenses. However, the estimation of the centroids is not examined and the proposed method is developed as a software solution only. In [16] sorting algorithms are compared which allow to increase the range of an SHWFS. The comparison has focused on aberrometry for the human eye, but the main results may be transferred to nearly every application predicated on wavefront sensing. Four different software-based approaches are compared: conventional, B-spline, Zernike and spiral algorithm. In [16], a 17×17 lenslet array has been examined with MATLAB but the MATLAB code has not yet been optimized for speed. The spiral algorithm based on [104] shows

twice the processing time of the conventional algorithm; while the processing time of the latter is too large already in the first place. On the other hand, the authors of [47, 83, 67] have shown that the processing time may be reduced drastically when hardware acceleration, i.e. an FPGA or other hardware implementations, is employed.

In [47], the dynamic range is extended without any hardware modification. In addition, improvements concerning the evaluation algorithms and several hardware changes have been suggested to the SHWFS to circumvent some of the limitations in the conventional approach [60, 59, 119]. These approaches, using astigmatic micro-lenses, exhibit the disadvantage that additional modifications to the regular SHWFS setup are needed and, among some of these approaches, cannot be scaled to a large and really fast SHWFS without degrading the performance. These hardware changes to extend the dynamic range or to simplify the evaluation should not be mistaken for hardware implementation of sorting algorithms i.e. [67, 47].

Another, often negligible fact is the problem of determining the position of each spot. In general, the assumption that locating the position of each spot is trivial (e.g. in [16]) does not hold when refraining from the conventional approach. Whenever latency and/or memory is critical or limited, however, the determination of the centroids may also become very challenging. Towards solving these issues, a solution for locating the spots has been developed during the dissertation and has been presented recently. It requires a minimal amount of time and memory. Additionally, it can be realized on an FPGA while being real-time capable [70, 67], see Sec. 3.3.2. These contributions also include a simple sorting algorithm (see Sec. 3.3.3) that provides a comparable small latency. However the solution shows a better performance in contrast to the conventional approach. Additionally, another advanced spot pattern ordering algorithm has been developed in [71], called modified spiral algorithm which is able to increase the range while preserving the low-latency of the previous contribution (see Sec. 3.3.4).

This research mainly contributes two solutions: Appropriate FPGA-based implementation such that latencies are minimized for real-time applications and development of a novel spot detecting algorithm when the spots leave their associated sub-aperture while preserving low-latency and real-time capability. The developed solutions demand less computational load when compared with conventional approaches, e.g. [47]. All these concepts have been implemented in software for validating the algorithms in MATLAB, allowing to easily compare the approaches with each other.

3.3.2 Connected component labeling (CCL)

CCL—also called connected-component analysis, region labeling, to name but a few—is an algorithmic application of graph theory. Subsets of connected components, often only called blobs, are uniquely labeled based on a predetermined heuristic, mostly using their neighbor relationship.

CCL is used in computer vision for detecting connected regions in digital binary

images. These binary images are determined after a thresholding step such that a pixel may only have a value of one or zero. CCL may also be used for road sign analysis to extract the signs out of a camera image stream and to alert the driver when he ignores a speed limit or to detect the lines of a road for a line assist system. There are two distinctively different approaches for CCL: single-pass and two-pass algorithms. The difference is that the two-pass algorithm needs saving the whole image which may result in a huge amount of required memory. The two-pass algorithm needs at least twice the time of the single-pass algorithm when being implemented sequentially.

As previously mentioned, the labeling used for the CCL is based on a predetermined heuristic; for this case, it is based on the 8-point neighborhood system. Popular neighborhood systems are the 4-point neighborhood system, shown in Fig. 3.5a and the 8-point neighborhood system shown in Fig. 3.5b, whereas the symbol s marks the actual pixel. The corresponding neighbors are marked in gray.

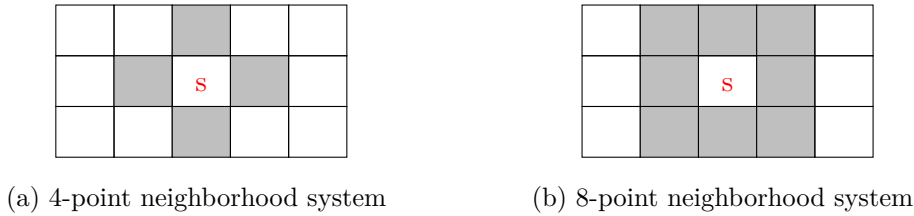


Figure 3.5: 4- and 8-point neighborhood system concerning pixel s

Let $\mathcal{C}(s)$ denote the pixelwise connected set of all iterated 8-point-neighbors starting from pixel s having a value of one for each pixel.⁴ Starting from this definition it is obvious that $\mathcal{C}(s)$ is a maximal set. Therefore whenever pixel r lies in $\mathcal{C}(s)$, then pixel s also lies in $\mathcal{C}(r)$ and vice versa. In other words, any of these r and s are iteratively connected by following their 8-point-neighbors accordingly. The associated binary image is determined by thresholding the digital pixel values which typically are 8-/10- or 12-bit intensity information. Thus, a binary image may e.g. look like Fig. 3.6. The thresholding step may be described as follows:

$$\text{pixelValue}(x,y) = \begin{cases} 1 & \text{for pixel}(x,y) > \delta \\ 0 & \text{otherwise} \end{cases}, \quad (3.20)$$

with δ as thresholding value. Parameter δ may be determined based on the image sensor and photon noise [118, 108] and by the ambient light when the experiment is not shaded.

When single-pass algorithms are used, a prevalent problem are so-called label collisions which happen when the situation is as shown in Fig. 3.7. Due to the **U** shaped object, instead of one identifier, two identifiers are used for labeling the blob. Therefore, at the position $?$ the two objects are getting connected but both are already labeled different from each other, in this case as '1' and '2'. Using an 8-point neigh-

⁴ If s is the middle pixel of a 3 by 3 pattern then an 8-point-neighbor of s is any of the 8 surrounding pixels that holds the same pixel value as s , see Fig. 3.5b.

	1	1	1	1		1		1				1		
	1	1	1			1		1				1		
	1	1	1	1		1	1	1		1	1	1	1	

Figure 3.6: Exemplary blobs sets in a binary image [71]

	1	1			2	2		
	1	1			2	2		
	1	1			2	2		
			1	?				

Figure 3.7: Label collision due to **U** shaped object [67]

borhood blob '1' and '2' are in fact one blob. Thus, both blobs should be merged into one. When using a 4-point neighborhood with the same situation as in Fig. 3.7, there would be no problem at all. The described situation is the so-called label collision.

Using single-pass algorithms we have to deal with the collision in parallel to the further processing. The existence of an **U** shape object at first sight is somewhat peculiar when labeling blobs of an SHWFS, in particular, since the distribution is usually gaussian for each spot and the lenses, which are used for building the lenslets, are convex. However, at the border of the blobs of an SHWFS, a label collision may occur due to the thresholding step, the pixel readout noise and a not ideal Gaussian intensity distribution of each spot. Hence, the resulting blob does not have to be a convex set after thresholding. In [13] an approach is presented which overcomes the problem of having label collision when using single-pass CCL while preserving strict real-time capabilities. The relabeling is realized in parallel to the further processing, thus, no latency or second run is needed.

For the SHWFS case, the area of individual blobs are not the desired properties itself. Instead, the centroids of the area of the blobs are important because these represent the tip-/tilt of the local wavefront over one lenslet. When examining the conventional approach (Equ. (3.16) and (3.17)), the centroids are also determined during application. For each labeled blob, a separate sum for the nominator as well as for the denominator of Equ. (3.16) and (3.17) is required. Due to the separate calculation for each blob, merging is trivial because it only requires to sum up the nominator and denominator separately. After finishing the CCL step, the consequence is that the centroids are not completely calculated because the division step of the sums are still missing. Therefore additional time is required to calculate the centroids

after finishing the CCL.

The CCL itself does not care about the size of the resulting blob. Additionally, the distance between blobs is unimportant. Each individual blob is identified based on the next neighbor-ship approach.

If the alignment of the SHWFS or of the other optical components of the setup is not optimal then so-called ghost spots or even double spots could occur. These spots have no corresponding lenslet, making the measurement invalid in general. Additionally, due to pixel noise and other aspects, spots may occur which do not correspond to a valid lenslet. In general, the intensity of the pixel corresponding to such a spot is usually lower than the valid spots and covers only a small area. Due to the mentioned characteristics it is sometimes desired to suppress blobs having less than a given value for the sum of the intensities, e.g. based on the value of the thresholding step.

One very simple, but effective method is to check how large the denominator is when calculating the centroids (Equ. (3.16) and (3.17)). The denominator is the sum of the intensity of the blob. If this sum is less than, say e.g. three times the global threshold value, it may be neglected. Due to this circumstance, blobs having only one or two pixels with intensities near the threshold value are disregarded efficiently; this helps to suppress ghost as well as double spots.

More sophisticated approaches may be found easily, but often are in conflict with the real-time requirement of the evaluation. Furthermore, a good initial setup, including calibration will also reduce such effects drastically.

In Sec. 3.4.1, the FPGA implementation for a single-pass CCL is presented which is somehow similar to the method in [13], but the operation mode of the CCL is implemented in a different way.

Generally speaking, the computation of the set $\mathcal{C}(s)$ is not a trivial task when to be implemented on a real-time system with minimal resources.

3.3.3 Straight line centroid ordering

After finishing the CCL and the corresponding step to calculate all centroids based on the valid blobs, the ordering of the spots to their corresponding lenslets is still unknown. Therefore the spot pattern has to be separated such that the mapping to the individual sub-apertures is achieved and valid. A simple centroid ordering algorithm is presented which resorts to a straight line separation.

The main idea of the straight line centroid ordering algorithm is that the spots of the lenslets are separated through a horizontal or vertical straight line. On account of this, the centroids are sorted regarding x- and y-coordinates, separately. Afterwards, a width is predefined based on the geometries of the SHWFS and the corresponding image sensor. Beginning with the first spot from the ordered list of centroids, in x- and y-direction individually, all spots within the predefined width are determined. The number of spots must be equal or less than number of lenses of the SHWFS in x-/y-direction. Fig. 3.8 shows exemplary the case for a 4×4 lenslet array. List. 3.1 introduces the pseudo code for this simple straight line centroid ordering algorithm.

The code has to be run for the ordered list of x- and y-centroids. The separators (horizontal and vertical) which divide the centroids are determined afterwards. The width variable in the pseudo code affects how large the distance between the first spot in a row and the following spots is allowed to be. In List. 3.1 the width is specified as 50% of the normal pixel width of a sub-aperture. If the next spot is more than ‘width’ away from the stored first spot in the row/column then the separator has been found. Otherwise, if the maximum number of spots in a row has been reached then the separator is also found. The value of the separator is the first spot in the row/column increased by the value of ‘width’. Graphically, separators may be interpreted as borders of the matrix where each box belongs to its corresponding lens.

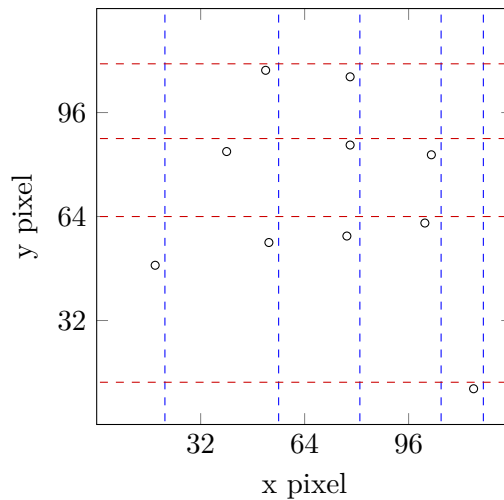


Figure 3.8: Example of the centroid ordering algorithm for a 4×4 example [67]

Another alternative to a predefined separation width would be to choose the width adaptively by calculating the difference between the maximal and minimal value of

Listing 3.1: pseudo code for simple straight centroid ordering [67]

```
1 width = ceil(width_x/max_row_sub-apertures*0.5);
2 cnt = 0;
3 counter_seg = 0;
4 for k in 0 to max_centroids loop
5   if( cnt == 0 ) then
6     cmp_value = centroid(k);
7     cnt = cnt + 1;
8   else
9     if( cnt == (max_row_sub-apertures - 1) ) then
10      msg("found 'cmp_value' as separator!");
11      counter_seg = counter_seg + 1;
12      cnt = 0;
13     elseif( (cmp_value + width) < centroid(k) OR
14             (cmp_value - width) > centroid(k) ) then
15      msg("found 'cmp_value' as separator!");
16      cmp_value = centroid(k);
17      counter_seg = counter_seg + 1;
18      cnt = 1;
19     else
20      cnt = cnt + 1;
21     end if;
22   end if;
23   if( k == (nb_of_max_spots - 1) ) then
24     if( cnt > 1 ) then
25       counter_seg = counter_seg + 1;
26     end if;
27     msg("found # 'counter_seg' separator!");
28     elseif( counter_seg == max_dim_matrix ) then
29       msg("error in centroid ordering!");
30     end if;
31 end loop;
```

the ordered list and dividing it by the number of sub-apertures per row/column, afterwards multiplying the value by an appropriate factor e.g. 0.5 for 50% of the average spot distance between the centroids.

Whenever one row or column of centroids is not present, e.g. due to the deformation of the wavefront, then the mapping turns out to be ambiguous. The problem may be solved e.g. by using a quad-optodiode because the sole error that occurs is a misalignment in x-/y- direction (tip- and tilt).

Note that the introduced ordering algorithm is just one of many possibilities. Clearly, if e.g. the number of centroids in a row/column is an integer value of power two then the algorithm may be simplified significantly. When the spots are misaligned such that the presented approach will fail, other classification methods may be used, as e.g. presented in Sec. 3.3.4. This may be the case, e.g. when the spots are aligned such that the shearing is huge. Then the proposed algorithm cannot work any more because it uses horizontal/vertical lines to separate the spots. Obvious modifications would be to use e.g. curvilinear separators since the separator need not necessarily be a horizontal/vertical line, however, the resulting complexity might rise excessively

such that a hardware implementation is impractical or even impossible when demanding real-time capabilities. The main advantage in comparison with the conventional approach is that e.g. the defocus of a wavefront is not limited anymore by the strict assignment of lenslets and area on the image sensor, see Fig. 3.9.

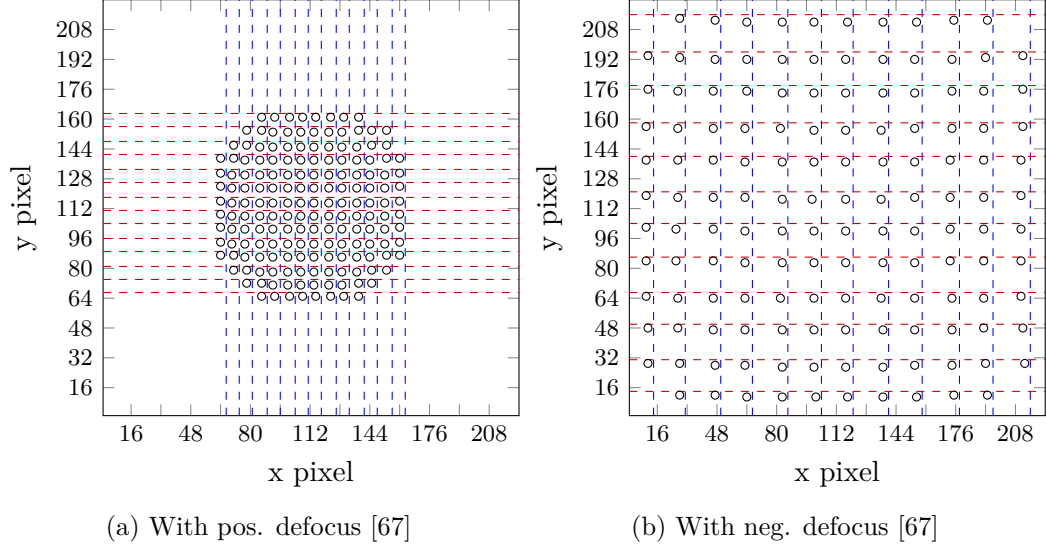


Figure 3.9: Segmented wavefront image after application of straight line ordering for the case of a huge positive or negative defocus

3.3.4 Spiral spot pattern adaption

The name ‘spiral method’ originates from the spiral-shape advancement during processing of the algorithm. The algorithm is based on local similarity and was originally developed for hexagonal spot patterns [104]. Fig. 3.10 is to exemplarily illustrate a spot numbering beginning with one. In the first step of the algorithm, the spot that is nearest to the ideal central spot is identified. For example, for a lenslet array with dimension 14×14 with a default width of 16 pixel per lenslet the ideal central spot would be located at pixel (104,104). The assumption, when starting in the middle of the wavefront to spiral, is that all arms will have the same length. Thus from the viewpoint of resource consumption and latency/run-time, this is an ideal initial point to start the spiraling.

25	10	11	12	13
24	9	2	3	14
23	8	1	4	15
22	7	6	5	16
21	20	19	18	17

Figure 3.10: Spiral algorithm numbering schema [71]

Searching the nearest spot with respect to the suggested position (104,104), the spiral motion is started. To this end, usually, a box is defined around the expected position and each previously found spot is checked whether the spot is located inside the defined box or not. If more than one spot position is found, the spot with minimum distance to the suggested position is chosen. Whenever no spot is found, the box size will be increased accordingly and a new search begins.

Starting from the first spot, the spiral algorithm proceeds. The spiral algorithm uses the coordinates of the previous spots for computing the expected distance for the next spot. Based on the actual spot coordinates and the expected distance to the following spot, the center-point of the next spot is defined. Thereafter, all points are examined if they are located inside the defined box with the afore-calculated center position. If multiple spots are found, then the closest spot, in the sense of least-squares, is chosen. In case that no spot is found, a dummy spot is assigned such that missing spots do not bother the progress of the algorithm.

The only exception from this sequence happens when determining the second spot: if no second spot is found at the beginning, the search area/box will be increased, followed by restarting the procedure. This exception is necessary, otherwise a huge defocus may let the algorithm fail immediately.

The search operation for the central spot has the disadvantage that in general $O(n)$ comparisons are needed (n specify the number of found spots). In view of the minimum distance (least-squares) search or the possibility of restarting the search

with increased box-size, retention of the algorithm may change and lead to a behavior that is difficult to predict, thus often-termed ‘non-deterministic’. Unknown and time-varying latencies are often incurred which is in conflict with the requirements in high-performance control. Due to worst case estimates, these latencies may lead to conservative controller designs that usually entail performance losses in the control system or drastically hinder the stability analysis of the controller.

The spiraling itself has a complexity of $O(n^2)$ since for any assignment, the inevitable search for the spot, which lays inside the defined box, requires $O(n)$ comparisons. Furthermore, if more than one spot lays inside the box, the search for the spot with the minimum distance may increase the complexity additionally by $O(n)$. Therefore, in total the complexity is $O(n^3)$ but usually, only a few spots are within the predefined area.

Adaptation

The spiral algorithm can be implemented in an FPGA with acceptable performance. But it is important to note that the implementation would not be good practice since, for example, a square shaped search area is used. As a side-effect, the implementation may show a non-deterministic run-time due to the cascaded structure when more than one spot is found after the initial search. Towards practicability, the spiral algorithm has been slightly modified to show a deterministic behavior, but still retains the prevalent beneficial characteristics of the original method.

The first step of finding the central spot was incorporated in the division after finishing the CCL which then calculates the centroids based on the previous labeled blobs. During this subsequent step, all centroids, one by one, are calculated. The centroid with minimum distance from the center position may be determined without any additional effort (regarding time without considering resources in the FPGA) through simple comparisons. Most importantly, no additional time effort is prevalent on account of this step. The problem of finding the central spot may be formulated as

$$x_c = \left(\frac{\text{img_width}}{2}, \frac{\text{img_height}}{2} \right) \quad (3.21)$$

$$\min_c \left((x_{c_x} - c_x)^2 + (x_{c_y} - c_y)^2 \right). \quad (3.22)$$

The feature of Equ. (3.22) is finding the centroid $c = (c_x, c_y)$ which is closest to the former calculated central point.

If the minimum distance between the central spot and the next spot (second spot) is too large, an error signal may be issued to display a badly conditioned image, e.g. when the defocus is too strong to measure the wavefront correctly. During initialization, the only decisive difference to the description in [16] is that the box is no longer a square, but a circle instead. This difference may be relativized when the arc tangent is incorporated in the distance calculation. Nevertheless, this would urge the need of

using e.g. a coordinate rotation digital computer (CORDIC)⁵ for evaluation of the arc tangent or a big look-up table with pre-calculated arc tangent values. Having in mind an FPGA implementation of the algorithm, the impact would be a huge effort during implementation of the algorithm. Additionally, a simulational study has shown that there is likely no benefit which justifies the overhead. The presented modifications i.e. getting rid of defining a square, searching the spots inside a given square and using the euclidean distance instead, leads to a deterministic behavior of the initialization while having a complexity of only $O(n)$. The complexity does not increase when calculating the centroids by the division step; therefore no additional latency arises.

Realizing the continuative classification of the spots wrt. the lenslet by finding the spot with minimum distance to the expected position also leads to a deterministic behavior of a subsequent eradication. If the distance is greater than a defined value then, instead of a spot, a pseudo spot is assigned to the lenslet which is the same behavior as shown by the original algorithm. In this way, even missing spots can be handled which may occur due to masking, absorption of parts of the wavefront or insufficient light intensity because of different reflection indices. Therefore, Gaussian shaped wavefronts are handled without any additional effort. The distance between neighbors may be calculated at any time. If no spot is present in the image then a pseudo spot position is used instead.

Simulation

In order to assess the performance of the spiral algorithm in comparison to the conventional algorithm, MATLAB simulations have been performed. These simulations were loosely based on the simulation study in [16]. The simulation was carried out with a 17×17 lenslet, each lenslet having a default area of 32×32 pixels on the image sensor chip to easily compare the results with the implementation of the spiral algorithm in [16]. Various kinds of disturbances, including entirely random disturbances, have been applied for revealing the limitations of the algorithm.

For a more realistic comparison of the results, analytically calculated slopes have been used. These are accessible as the real wavefront is known in advance. This circumstance yields to a fair comparison of the results from the implementation of the algorithm in MATLAB. The number of erroneously assigned spots are determined as well as the RMS error of the reconstructed wavefront. For the wavefront reconstruction, the slopes are fitted based on Zernike polynomials of 6th order, which corresponds to order 28 of the corresponding polynomial (see Sec. 3.1 for specific properties of the Zernike polynomials).

Fig. 3.11 represents a simulated wavefront where the camera sensor values are marked in white and the lenslets from the SHWFS in red. In the simulation example the disturbance is chosen to be in the usual range. As shown in Fig. 3.12, the conventional algorithm fails to assign all spots to its lenslets correctly. The result is an RMS

⁵ CORDIC, often known as digit-by-digit method and Volder's algorithm. CORDIC is a simple and efficient algorithm to calculate e.g. hyperbolic and trigonometric functions.

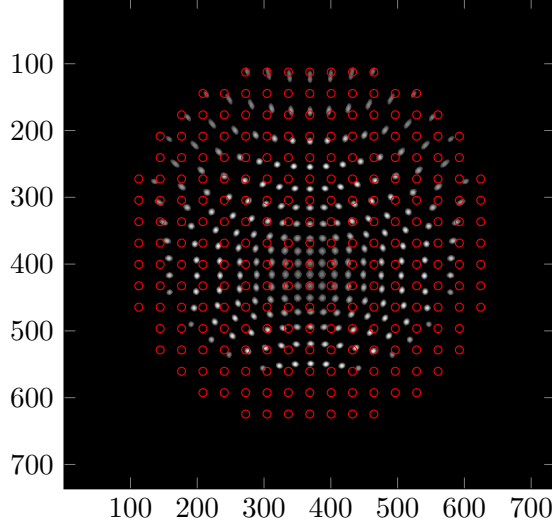


Figure 3.11: Simulated wavefront image and associated lenses, colored in red [71]

error of about 69.8 % which consequently leads to incorrect reconstructed wavefront (see Fig. 3.14a).

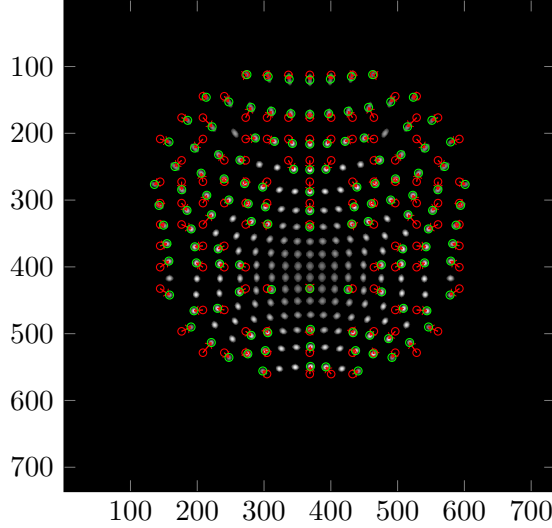


Figure 3.12: Spot ordering of the wavefront given in Fig. 3.11 using the conventional algorithm: 72 spots failed, the RMS error of the wavefront is in total $2.093 \approx 69.8 \%$ [71]

Fig. 3.13 instead shows the results when the spiral algorithm is applied. All spots are properly assigned to the corresponding lenslets and only a small RMS error of 2.9 % is observed. This error is due to the Zernike-polynomial based finite reconstruction and to numerical errors during the calculation of the centroids such as the modeled pixel noise. Tip/tilt is not incorporated in calculating the RMS error. For being sure to have correct tip/tilt information, the utilization of a quad-optodiode is recommended as the spiral algorithm, regardless of modified or original, does not ensure that the tip/tilt-information of the wavefront is correct. This limitation is the same as using the straight line ordering algorithm (see Sec. 3.3.3). Fig. 3.14b shows the

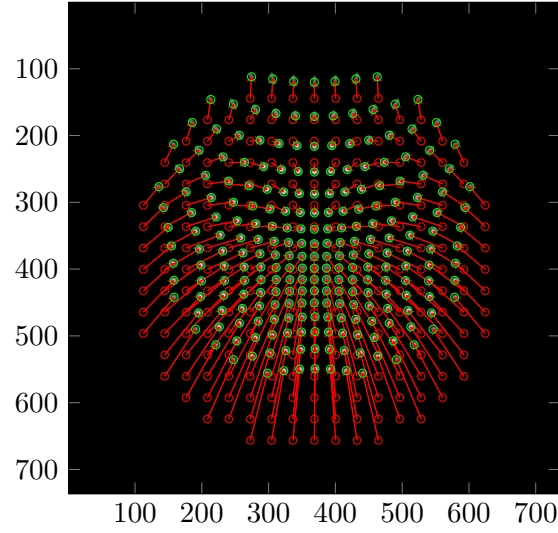


Figure 3.13: Spot ordering of the wavefront given in Fig. 3.11 using the spiral algorithm: no spots failed, the RMS difference error of the wavefront is in total $0.086 \approx 2.9\%$ [71]

corresponding, correctly reconstructed wavefront based on the allocated spots using the modified spiral algorithm having identical results as the original spiral algorithm.

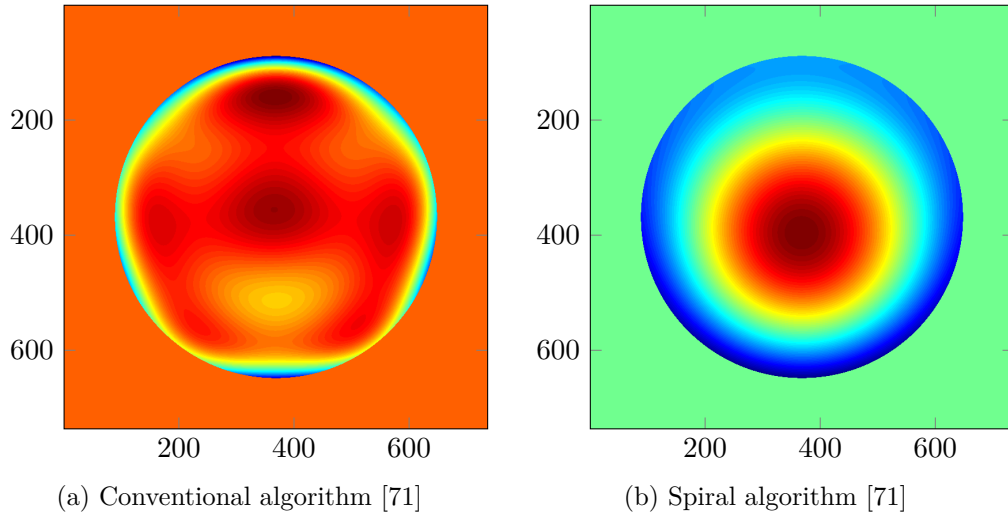


Figure 3.14: Reconstructed wavefront based on application of conventional (Fig. 3.14a) and spiral algorithm (Fig. 3.14b) for determining spots

3.3.5 Adaptive thresholding

During several experimental tests, it has been observed that the intensity of the complete laser beam, therefore of the spots as well, is not as uniform as desired (see Fig. 3.15a). A typical laser beam has an intensity profile of Gaussian shape. The spots in the center of the laser beam have a higher overall intensity than the spots near the corner. Usually, this is not a problem because by adopting the threshold to a low value, see Equ. (3.20) for the threshold definition, nearly all of the available spots can be detected reliably. However, it is not surprising that the threshold value cannot be decreased without incurring limitations.

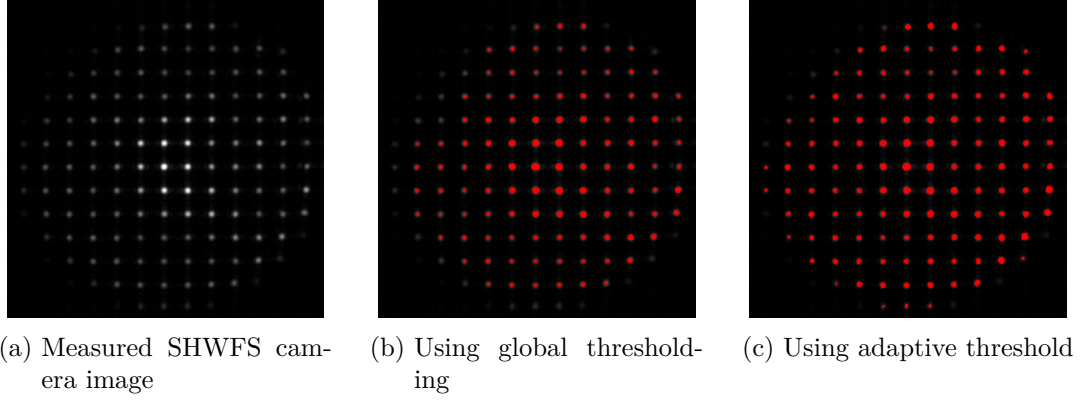


Figure 3.15: Difference between global and adaptive thresholding when Gaussian intensity profile is prevalent [65]

One reason is that due to the typically rectangular aperture of each lens, special diffraction patterns will occur, see Appendix A.1. The intensity profile of each spot, resulting from the typically rectangular lens for simplicity, is not only a dot. Instead, it appears as a dot having a cross as an overlay in x- and y-direction. This cross may have nearly the same intensity for spots in the center of the image as the maximum intensity of the outer spots due to the global Gaussian intensity profile. In this case, a problem with the CCL is likely to happen because more spots or even spots in direct local neighborhood of a spot will be detected.

The problem of not finding all spots is actually a technical one. It is obvious that there are cases when not all spots can be found at all. Nevertheless, some challenging problems can occur when the spots at the border are not detected, as it is the case in Fig. 3.15b. For the calculation of slopes based on the ordered spots, a reference is required for each spot such that the difference between the spot position can be calculated. Ideally, the reference is the default position having a plane wave (see Sec. 3.2). In spite of this, it is not advantageous since either the relative deflection is measured or the relationship between the ordered spots and the ideal position is missing, see Sec. 3.3.3. As discussed during the presentation of the two ordering algorithms, the straight line segmentation approach is not able to measure absolute tip/tilt values as the spiral algorithm. Fig. 3.15b shows the case when not all spots could be found when applying global thresholding. During recording the actuator

influence function (AIF), sometimes, the spots at the left/bottom corner are also detected because their intensity will be randomly higher than the thresholding value. Thus, the number of lines and rows will change and the assignment to the lenslet array will change due to the behavior of the algorithm. The assignment can be shifted either up/down or left/right when one row/column will change.

Therefore, it is advisable to improve the thresholding step such that ideally, all spots are detected correctly at every captured frame. This may be seen as an unrealistic assignment, but every little improvement is likely to help quite a lot to minimize possible issues. In the field of image processing, several techniques to improve the global thresholding have been developed. Most of them are not suitable for this specific task because the underlying problem is a non-uniform background. There is a global intensity difference from the center to the border while having in the center disturbances with the same intensity as the spots at the border. Also, the intensity maximum may not lay in the center of the image if the adjustment of the optical setup is not very precise. But still, the main problem remains the same as this circumstance is a shift of the center of the intensity profile.

To improve or at least partially solve the problem, the idea is to approximate the inverse of the intensity beam profile and adopting the threshold value depending on the position of the received pixel of the image frame. As it is known that the beam is typically Gaussian, its inverse should be computed. This inverse is quite complex for calculations on an FPGA without special arithmetic. Therefore decreasing the threshold value depending on the euclidian distance from a given/determined center can be used as a rough approximation for the inverse, i.e. the expression

$$\text{ceil} \left(\left((X - \text{centX})^2 + (Y - \text{centY})^2 \right) \cdot \text{scalVal} / 2^{16} \right) \quad (3.23)$$

where X and Y denotes the actual pixel position, centX and centY are the assumed position of the maximum intensity. For calculating the euclidean distance, the square root operation is required. The value of expression (3.23) is subtracted from the global threshold value such that the new threshold value is similar to a cone instead a two-dimensional parable. The slope is adjusted by changing scalVal , i.e. an eight bit integer. The result is scaled by 2^{16} such that values in the range from 0 to $\frac{1}{255}$ can be achieved. Fig. 3.15c depicts the result. The calculation may easily be pipelined for implementation on an FPGA such that the step only takes one cycle. Furthermore, only an addition, subtraction, multiplication and shifts are used; no square-root or even a division has to be calculated. Therefore, only one cycle delay arises without requiring large memory capacity and special efforts for the implementation. The critical part still is the CCL implementation due to its complexity.

3.3.6 Repositioning of centroids

Another problem is the ordering of the determined centroids into the corresponding matrix which has the dimension of the lenslet array (see Sec. 3.3.5). The positioning is ambiguous when the number of rows and columns are not identical to the dimension of the lenslet array. Nevertheless, this will only influence tip and tilt which can be measured separately.

In fact the problem arise when the dimension of the segmented centroids are not the same as the original number of lenslets in x- and y-direction. Then, the assignment can be shifted either up/down or left/right when one row/column will change.

If we have twelve rows and twelve columns but a lenslet array of 14×14 , it is not well-defined how these centroids must be placed inside the resulting matrix. That is, the placing is ambiguous. If the positioning is incorrect, errors will occur when the slopes are calculated due to the wrong assignment. Therefore the result will be an invalid subtraction between the reference centroids and the actual one. But by means of some external quad-photodiode with external tip/tilt compensation, additional information is available for guessing the right positioning. Another possibility is to neglect the real tip/tilt and place the centroids, e.g. centric, inside the matrix as it has been realized in the introduced ordering implementation.

Of course, this does not solve the ambiguity but helps to measure other aberrations except for absolute tip/tilt values.

Table 3.2: Truth table of the logical **AND** operator

p	q	p AND q
0	0	0
1	0	0
0	1	0
1	1	1

One chance to prevent the possible shift is exploiting the specific shape of the centroids, typically a circle (see Fig. 3.15a). Some kind of convolution may be applied to determine the correct shift, if any has appeared. Theoretical considerations show that the typical shift is either $+1/0/-1$ in each direction (x/y), resulting in maximum nine possibilities, see Fig. 3.16. It is unlikely that two rows or columns will appear at once. But in this case the proposed algorithm will tend to fail.

The reference for e.g. calculating the slopes has been taken exemplarily in Fig. 3.17 and is marked with x . The algorithm for placing the determined centroids centric in the matrix will position the values as visualized in Fig. 3.17. In this case, the resulting matrix will have dimension six; three rows and columns have been found. Therefore three rows and columns are empty such that the algorithm divides three by two without rest. Thus, the result is equal to one, resulting in an empty row and column at the beginning and two empty rows/columns at the end.

At some time instance t , the evaluation will detect the spots marked with o as

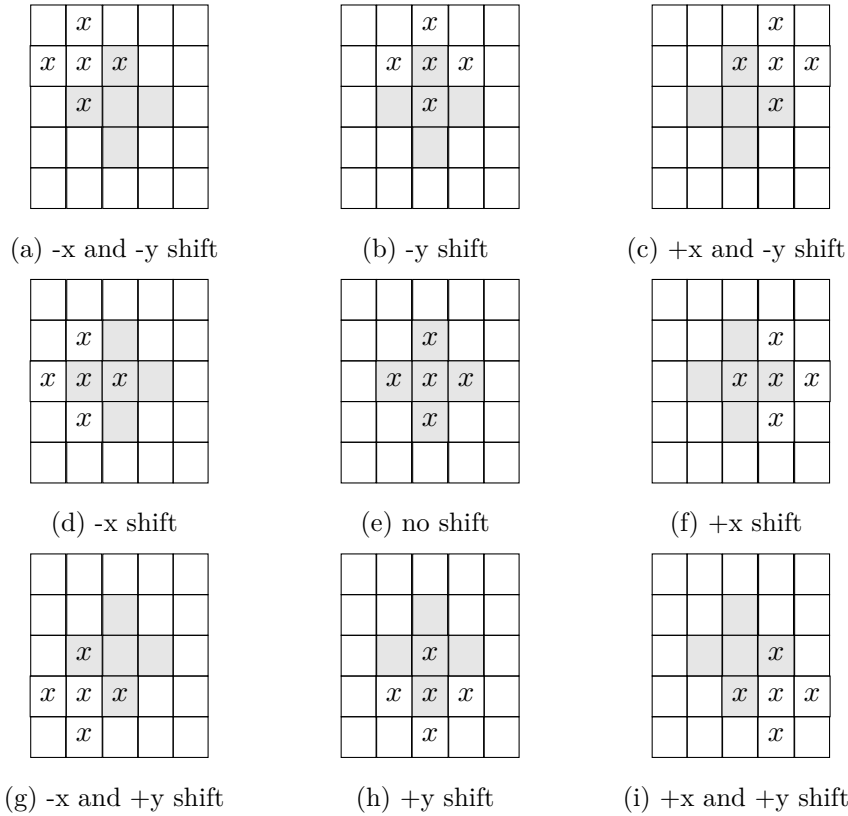


Figure 3.16: Visualization of the nine different possibilities of typically occurring shifts

well; often occurring during the capturing of the AIF. As the actuators are driven with the maximal amplitude, a maximum deflection is achieved. Therefore, this is likely to happen during closed-loop operation, i.e. when controlling the system while having large aberrations. The result is that the centric positioning of the matrix will lead to the case, as marked with the gray background because four rows and columns have been determined. Thus, also one empty row and column will be inserted by the centric positioning algorithm. The newly determined spots will lead to the case such that the assignment of the reference and the determined ordered matrix is no longer valid. One method to detect the shift is comparing the shifted reference with the actual matrix by application of the logical ‘AND’ operator, see Tab. 3.2 for the definition of the operator.

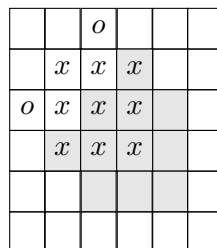


Figure 3.17: Centric positioned centroids and the resulting positioning error when new spots (o) have been detected [64]

The new determined matrix will be shifted in all nine possible directions (see Fig. 3.16) and associated with the original (reference) matrix by application of the ‘AND’ operator. The result will be a binary matrix having only entries of ‘0’ and ‘1’. For each shift, the number of ones inside the binary matrix will be counted. Depending on the number of ones inside the matrix, the most likely shifting based on the result of the nine sums, is determined. In the case of Fig. 3.17, without shifting, we would have six ones in the binary matrix. Thus, the sum of ones of the binary matrix will be six. Whereas when shifting -x and -y (shown in Fig. 3.16a), the number of ones would be nine. In the given case, it is very easy to check that this is the maximum number of ones for all nine different possible shifts. Therefore, the matrix is repositioned by -x and -y such that the match is maximized.

The presented approach is just one of several approaches, e.g. another approach requires some hardware modification and is similar to [59, 60]. When one or more lenses inside the lenslet array are masked or some lens generate a special shape of the spot this information can be used to determine the correct positioning. In the case of the given SHWFS, this is not possible because the SHWFS uses an regular lenslet array which does not have any integrated marker.

There even is the possibility that the former determined spot assignment is compared with the actual position. Then, some kind of tracing as e.g. in [116] is done, but several problems are prevalent with such a solution [67]. Additionally, tracing is not really suited for implementation on an FPGA because it requires storing a lot of individual information and typically leads to long logic paths. A long logic path will likely result in a critical path, thus, will limit the maximum achievable clocking frequency.

3.4 FPGA Algorithm Implementation

Sec. 3.3 treats the different algorithms for determining the centroids as well as the corresponding ordering of centroids to their lenslets whereas this section focuses on the implementation of the algorithms by using VHDL⁶.

As FPGA target, a Xilinx Kintex-7 FPGA has been used which is the heart of the peripheral component interconnect express (PCIe) FPGA card mounted in the real-time system (see Sec. 5.1).

3.4.1 CCL for SHWFS's

The basic principle of CCL has been treated in Sec. 3.3.2. When using CCL for an SHWFS to evaluate the centroids, an 8-point neighborhood is used. The neighbored pixels are named as depicted in Fig. 3.18. The label D , visible in Fig. 3.18, is only

	A	B	C	D	
	E	?			

Figure 3.18: Labeling of the pixel neighbors for the CCL FPGA implementation [67]

marked due to the later implementation and does not increase the neighborhood. Furthermore, Fig. 3.18 differs from Fig. 3.5b as only the upper and left pixel of the actual positions are used because the single-pass approach is applied.

row 1	1	2	3	4
row 2	5	6	7	8
row 3	9	10	11	12
row 4	13	14	15	16

Figure 3.19: Construction of the pixel stream of the transmitted image when using a 4×4 pixel image sensor

Since the pixels are streamed sequentially and start in the top left corner from left to right (see Fig. 3.19), only pixels above and left of the pixel ? (Fig. 3.18) are required for determining the connected components, usually called blobs.

Fig. 3.20 shows the block diagram of the CCL implementation in VHDL. As shown in Fig. 3.20, the CCL consists of four different modules. First, there is the CCL itself to whom a row register is connected. This row register stores one line of the former pixel to decide whether the pixel are connected or not. The label stack provides new labels but can also handle label collision when a label is freed due to a occurring label

⁶ VHDL is a hardware description language which is used in electronic design automation to describe digital and mixed-signal systems, i.e. FPGAs.

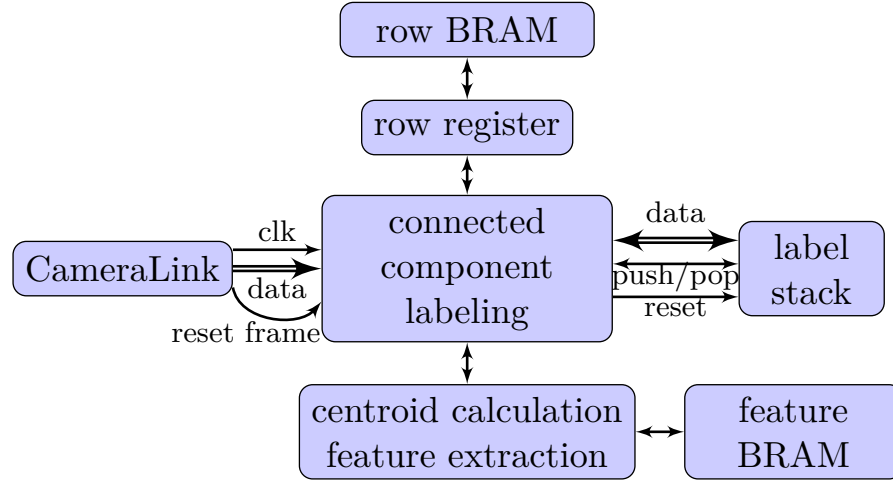


Figure 3.20: Block diagram of the complete CCL FPGA implementation [67]

collision. The third module is the so-called centroid calculation or feature extraction, where the nominator and denominator are stored for later calculation of the centroids.

The row register is divided into four parts. The register is loaded with the old label data from the former line automatically such that the connected-component could always read A , B , C and E (see labeling in Fig. 3.18). The label stack is implemented with flip-flops (FFs) and not with a block random access memory (BRAM)⁷ because it is necessary to reset the label stack to its original state when a new frame begins. Nevertheless when enough time between images is given, the label stack could also be implemented with BRAM as with a second read/write port. The content of the BRAM could be restored in the meantime instead of using a dedicated reset.

Another solution is the use of two BRAMs and switch between both at run-time. When odd frames are processed, the second BRAM could be re-initialized with a counter, so when a even frame starts the just re-initialized second BRAM is used and vice versa. Due to the one cycle clock latency of BRAMs, pixel D is set as read address during the processing of pixel $?$. This is realized to be able to access the centroid data, if pixel D has an assigned label.

When a label collision happens, the left label is used and the other label is pushed back to the stack such that this label can be again used. In parallel, the content of the other label in the *centroid calculation/feature extraction* is merged with the content of the left label. Because the left label has already been used, there is no need to relabel items left of the actual position in the row register.

In a label valid register, the entry of the other label is marked as invalid. Therefore, the label which is pushed back onto the stack, may be reused if necessary. Thus, even when a lot of collisions occur the number of required labels are minimized.

Merging two labeled blobs is possible because the division, see Equ. (3.16) and (3.17), is initiated after finishing the processing of the complete frame. Another

⁷ BRAM means Block RAM and is a dedicated RAM inside the FPGA to save efficiently data. BRAM could only be accessed sequentially and have one clock latency.

possibility would be to start the centroids calculation for each blob after e.g. y is greater than the minimum y value of the centroid plus a determined constant. This would use the circumstance that a spot could not be greater than a defined value while still measuring the wavefront with the SHWFS correctly. After processing the complete image, the valid entries in the BRAM are marked through a separate label valid register for further processing.

The overall number of labels is determined by the depth of the stack. The use of lookup table (LUT)s and FFs is minimized through utilizing BRAMs for storing the previous line except for the *row register*, which has a minimum width of only 16 times the logarithm of the number of labels.

For obtaining the centroids, a subsequent division step has to be performed after finishing the CCL. Therefore, the Xilinx Divider Generator IP-core is used to perform the division with a predefined number of fractional digits. Based on the data valid register, the feature BRAM entries are accessed. If the corresponding bit in the register is valid, the division is started. Thus, in total, the maximal number of divisions are limited to the maximum number of blobs (n), so the complexity is $O(n)$. Each division step requires a specified amount of clock cycles which is determined by the width of the divisor and dividend. Of course, the number of fractional digits influence the amount of required clock cycles also. Through modifying the pipeline mode of the IP-core itself, a tradeoff between resource consumption as well as maximum clocking speed can be realized.

3.4.2 Straight line centroid ordering

Implementation of the single-pass CCL with the straight line centroid ordering has been realized with VHDL. As synthesis tool, ISE WebPACK Design Software 13.4 from Xilinx was used for the first attempt. The original implementation was initially optimized for the Xilinx Spartan-6 FPGA series [67].

Finally, the implementation has been ported to the Kintex-7 FPGA. The following results e.g. the resource consumption, has been analyzed when using a Kintex-7 FPGA.

Centroid Ordering

For applying the straight line centroid ordering algorithm, a separate ordered list regarding x- and y-value is required. The ordering based on the x- and y-coordinates is implemented in hardware, without using a soft-core, due to performance aspects as well as simplicity. For the ordering problem a lot of different approaches exist. Due to the overall size of $n = 256$ when using the HASOTM3 Fast SHWFS, a trade-off between performance and space complexity has to be disposed. The HASOTM3 Fast SHWFS has 14×14 sub-apertures and the next power of two is 256; therefore n is equal to 256.

The ‘selection sort’ has a space complexity of $O(1)$ but with worst case performance $O(n^2)$. The ‘Batcher odd-even mergesort’ has a performance of $O(\log^2(n))$ but with a space complexity of $O(n \log^2(n))$, see [21].

In view of the large size of n , a trade-off between ‘selection sort’ and ‘Batcher odd-even mergesort’ is preferred. In order to preserve the use of FFs, the centroid coordinates are stored in a BRAM. A second BRAM is used for saving the ordered values. The n data in the first BRAM are read sequentially. By comparison of each value, the minimum and maximum value is determined. For improving the performance further, it is also possible to find m max/min values in parallel whereas m is the number of parallel comparisons.

Nevertheless the complexity for comparing the values rises exponentially with m . The performance is $O(n \frac{n}{2^m})$, which also is $O(n^2)$, but, thanks to an adjustable pre-factor, it may be much faster. The memory consumption increases to $O(n)$ due to the use of the second BRAM. In our case, a value of $m = 3$ results in a good trade-off between space complexity and performance.

With this setting, the implementation inside an FPGA is efficient since only $2m$ registers are used for temporary saving the min/max values. Of course, some additional logic for comparing the registers is necessary. The BRAMs grant a clocking frequency which in general is near to the maximum of the FPGA. The maximum combinatorial path is very short. Hence, the latencies due to routing are minimized.

Division

For the division step, which is needed to calculate the centroids after the CCL, the ‘divider generator IP core’ from Xilinx has been used with a configured latency of 16 cycles. The division is implemented as High-Radix division due to the required bit length. Through these settings the core is able to operate with a frequency above 80 MHz which is required to cope with the pixel stream of the HASOTM3 Fast SHWFS without any intermediate storage.

Hardware Specific Results

The first version of the straight line ordering approach has been realized with the FPGA board based on the TE0600 micro-module from ‘trenz electronic’. This micro-module employs a Spartan-6 150LX Speedgrade 3 FPGA from Xilinx. For this experiment, the baseboard shown in Fig. 3.21 has been developed. The baseboard is an in-house developed four layer board with Gigabit Ethernet, a FT2232H USB2.0 FTDI chip and an universal serial bus (USB) to universal asynchronous receiver/transmitter (UART) bridge. Using the USB port, specific test images have been loaded into the DDR-SDRAM of the FPGA to examine whether the results are as expected.

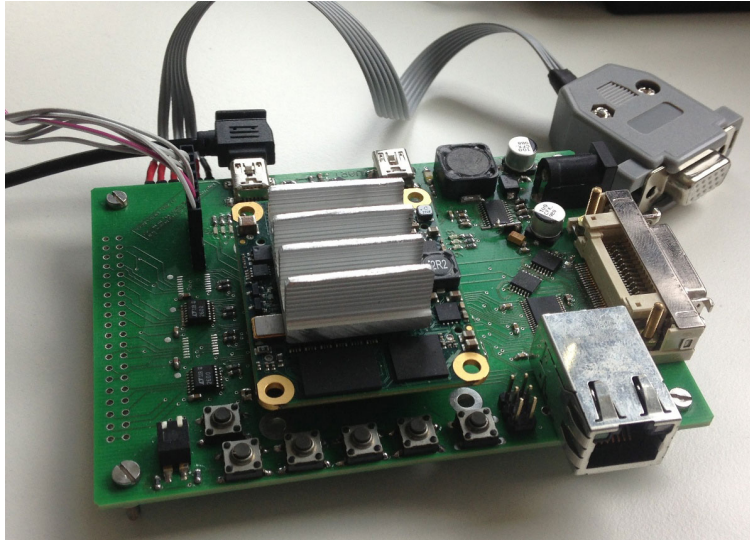


Figure 3.21: Custom build Spartan-6 FPGA board with CameraLink interface

The proposed algorithm for the segmentation, shown in List. 3.1, has a worst case performance of $O(n)$ for a single run. Due to the segmentation in x- and y-direction, either two instances are used in parallel or $2n$ clock cycles are required.

The WFS used for experimental testing is a HASOTM3 Fast from Imagine Optics which is based on the Bobcat camera ICL-B0620M; for more information see Appendix A.9. The SHWFS is connected through a CameraLink interface with either an external deserializer DS90CR288A from Texas Instruments or using the integrated SERDES⁸ of the FPGA. Additionally by using a video graphics array (VGA)-port

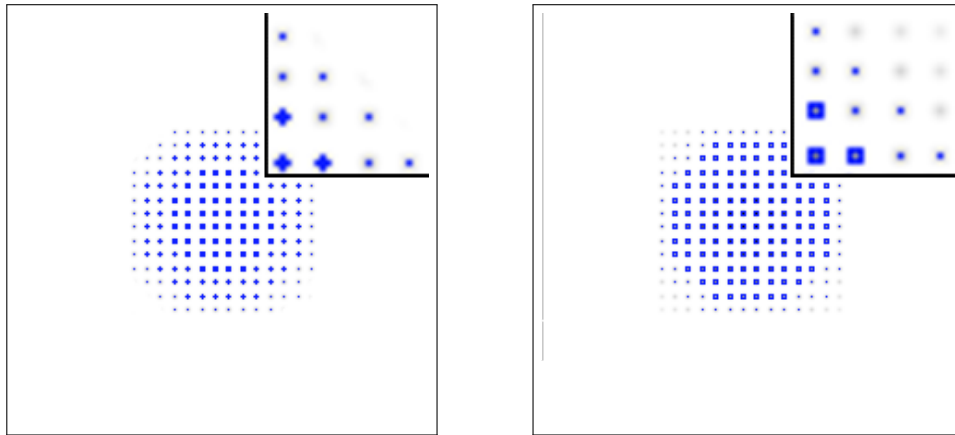
⁸ SERDES are serializer/deserializer integrated in the FPGA which convert data between serial data

which is connected to the FPGA, the original camera image, the determined centroids and the horizontal/vertical lines are displays via VGA such that only a conventional computer monitor is required for having a visualization of the algorithm.

The pixel clock is 40 MHz and either single-tap or dual-tap may be used to transfer data over the CameraLink interface, mostly depending on the desired frame rate of the SHWFS. Consequently, the CCL must be able to operate at least with a frequency of 40 MHz in single-tap mode. If the pixels are to be processed sequentially in dual-tap mode then the CCL as well as all subsequent modules must operate at least with a frequency of 80 MHz in order to be able to process the pixels without storing them. The camera settings are adjustable via the *Bobcat Configurator* by means of forwarding the serial port, transmitted over the CameraLink interface, to the personal computer (PC) through the integrated USB to UART bridge on the FPGA board.

If the achievable maximum clocking frequency of the CCL does not surpass the pixel clock then BRAMs may be used to store the values at full-speed. With the second read-port they may then be accessed with a slower clock to simplify timing. Of course, the latency will rise due to the slower read-out clock (quotient of the camera clock divided by the read-out clock) when compared with the nominal case.

A series of test images and experimentally measured wavefronts from an optical test setup serves for validating the presented approach. A thresholded test image is shown in Fig. 3.22a which represents a strong defocus. Fig. 3.22b shows the resulting image where the detected spots are marked with a bounding box in blue.



(a) Thresholded marked blue, partially zoomed (b) Blue bounding box around detected blobs, partially zoomed

Figure 3.22: Exemplary test image based on VHDL simulations using testbenches

Fig. 3.9a shows how the algorithm separates the detected spots in the sub-apertures. The separators are drawn in dashed lines. The default sub-apertures border is at a multiple of 16 pixel because the width/height of the image is 224 pixel and 14×14 lenses/sub-apertures are used. As long as the wavefront does not result in an enormous shearing, the proposed segmentation algorithm works reliable. In comparison with

and parallel interfaces in each direction.

the conventional algorithm (shown in Sec. 3.2), the proposed algorithm will always perform better than the standard one. As long as the width is 50% of the normal pixel width of a sub-aperture, see Sec. 3.3.3, both algorithms behave more or less the same. Nevertheless the proposed algorithm is able to handle wavefronts with a much stronger defocus than the conventional approach. Its performance may be further increased when choosing the width adaptively (as suggested in Section 3.3.3).

As mentioned in Sec. 3.3.3, the proposed segmentation algorithm is well-suited for a strong defocus, e.g. pronounced in laser material processing. This is the target application of this first and simple approach. Fig. 3.9b depicts a situation where the wavefront does not only have a defocus.

The device utilization for the detection of the spots (by using CCL), calculation of the centroids, ordering the centroids and segmentation of the spots is shown in Tab. 3.3 including LUTs, register and other peripheral consumption. The centroid position is calculated with an accuracy of four fractional digits. The values in Tab. 3.3 and 3.4 are to be understood without the CameraLink module, also neglecting the possible necessary BRAMs, due to the maximum clock frequency, if required.

Table 3.3: FPGA logic consumption with $n = 256$, $m = 3$, image width/height 224 pixels (Xilinx Spartan-6 XC6SLX150-3FGG484 FPGA) [67]

Logic Utilization	Used	Available	Util.
# Slice Registers	5703	184304	3%
# Slice LUTs	5966	92152	6%
# fully used LUT-FF pairs	2248	9421	23%
# Block RAM/FIFO	9	268	3%
# DSP48A1s	20	180	11%

Table 3.4: FPGA logic consumption with $n = 256$, $m = 3$, image width/height 224 pixels (Xilinx Kintex-7 XC7K160T-2FBG676 FPGA)

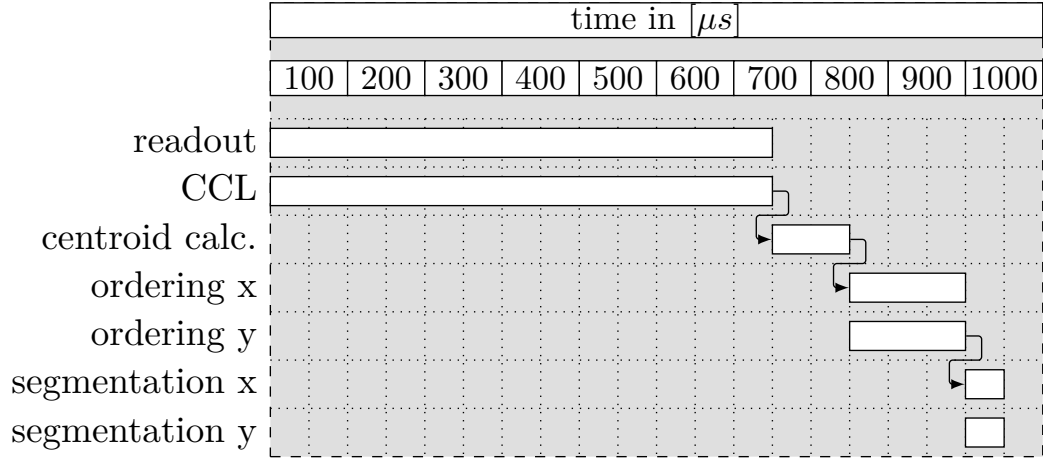
Logic Utilization	Used	Available	Util.
# Slice Registers	4553	202800	2%
# Slice LUTs	6238	101400	6%
# fully used LUT-FF pairs	2186	8605	25%
# Block RAM/FIFO	6	325	1%
# DSP48E1s	20	600	3%

The minimum clock period when using a Spartan-6 is 12.300 ns which correlates to a maximum frequency of 81.299 MHz. Using a Kintex-7 FPGA, the maximum frequency rises, i.e. 6.410 ns correlates to a maximum frequency of 156.006 MHz. The required time for executing the complete series of steps in our experimental implementation is listed in Tab. 3.5. The sequential processing as well as a graphical representation of the time and each individual step is shown in Fig. 3.23.

It is worth mentioning that the maximum frequency is limited by the longest combinatorial path which is defined by the CCL module. The result is that the frac-

Table 3.5: Maximum required clock cycles with $n = 256$, $m = 3$ and image width/height 224 pixels [67]

sequence	# cycles	clk [Mhz]
read img (w/o syncing cyc.)	50176	80
division centroids	4352	80
sort centroids BRAM	10923	80
segmentation centroids	512	80

Figure 3.23: Time evolution with 80 MHz, see Tab. 3.5, (each step rounded up to $50 \mu s$) during image processing

tional bits can be increased without degrading the maximum frequency, even without increasing the time-delay. Of course, one consequence is that the logic utilization increases when more fractional bits are required.

3.4.3 Modified spiral algorithm

The overall complexity of the modified spiral algorithm is $O(n^2)$. Each minimum distance search needs $O(n)$ comparisons, with two multiplications and one addition for each centroid. Having a constant run-time when using the modified spiral algorithm is one advantage, but also the overall complexity is reduced from the former one, which was $O(n^3)$. The only major difference in the behavior of the modified spiral algorithm compared with the original algorithm is regarding the second spot. The modified spiral algorithms does not rely on moving a box when no spot is found in the defined area. Thus, special care has to be taken during implementation to assign the first spot reliable.

The centroids are usually stored in BRAMs in order to use the prevalent dedicated hardware of the FPGA. Therefore, the centroids can only be accessed sequentially. When using two individual BRAMs for storing the centroids, a reduction of the time needed for accessing all centroids to half of the previous time is possible because two individual read/write ports may be employed. Nevertheless, the complexity itself is still $O(n)$. Parameter $m = 2$ sets that two BRAMs are used for the storing the centroids. This is a good trade-off regarding logic and space consumption as well as for the performance. With $m = 2$, due to the two BRAMs, two parallel searches having reverse order for finding the minimum distance are employed. On the other hand, this increases the number of multipliers and adders by factor m but the benefit is that the run-time is half of the previous. Nevertheless, the same applies here as in [67], namely that the individually found values have to be compared with each other and the complexity arise exponentially with m .

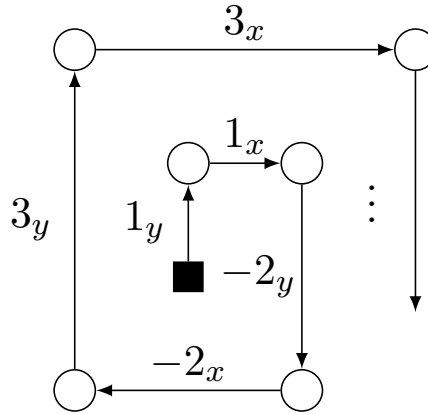


Figure 3.24: Spiral movement with distance ordering [71]

Fig. 3.24 visualizes how the spiraling algorithm determines when the arm has to be rotated by 90° clock-wise. The distance is the same for the x/y step, e.g. 1_x means that the arm is (usually) one aperture width away in the x-direction. Universally, two arms have the same length with different x - or y -directions. Both next arms may be calculated by taking the absolute value increased by one and flipping the

sign of the value wrt. the previous sign. Whenever previously found centroids are available for adjusting the distance accordingly, these widths are used instead of the fixed predefined aperture width. Automatically adjusting the distance by using the distance between neighbors allows to cope with heavy deformations such as a strong defocus. Starting in the middle of the image, the deformation is usually small in a given local region, but is getting larger and larger when moving outside the region. Through the adaptation, this behavior is considered accordingly and does not degrade the performance.

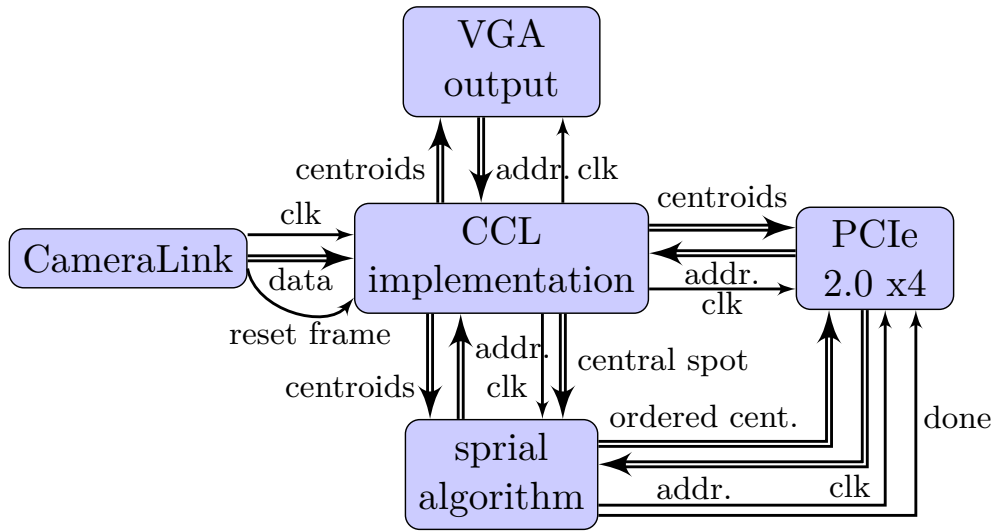


Figure 3.25: Simplified overview of the FPGA implementation for the modified spiral algorithm [71]

Fig. 3.25 gives an overview of the implemented modules in terms of a block diagram. The CameraLink module receives the data from the SHWFS, the CCL implementation labels all blobs. Thereafter, the centroids are calculated based on the blobs. The centroids are passed to the VGA output for visualization on a normal computer monitor. Furthermore, they may be transmitted to the computer in real-time by using the PCIe interface. A controller may then use the determined and already segmented centroids for calculating appropriate signals to drive the actuators, e.g. of a deformable mirror (DM). Announcing the end of the computation by a ‘done’ signal, the time needed for evaluating the wavefront may be determined easily. This time corresponds to the latency which is additionally introduced by evaluation of the wavefront.

Saving the identified spots in BRAMs reduces routing latencies and space but the content may only be accessed sequentially with one clock latency. Therefore two additional steps are required to determine the distance between the two neighbors at each time. Consequentially, when the SHWFS consists of a 14×14 lens array, each spot requires to perform $14 \cdot 14 + 2$ steps.

Results

The sequence and time evolution, beginning with capturing the image until the finish of spiral algorithm execution, is depicted in Fig. 3.26. These timing information is valid for $m = 1$, $n = 256$ and a 16×16 lenslet array having a clocking frequency of 80 MHz. Analyzing Fig. 3.26, it is clear that almost half of the time is needed for

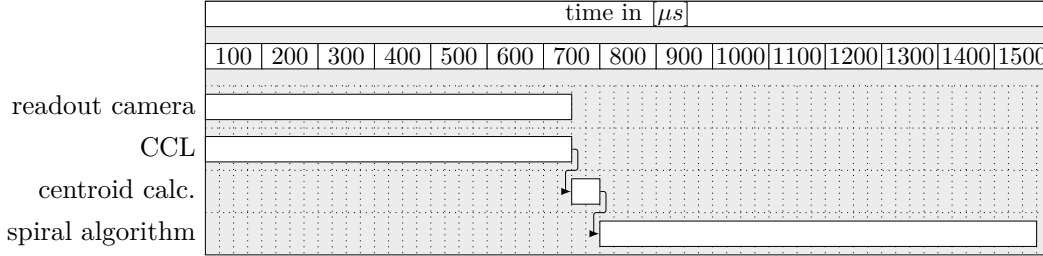


Figure 3.26: Time evolution of the complete processing where each step is rounded up to $25 \mu s$ for the modified spiral algorithm [71]

transferring the image from the camera into the FPGA. Additionally, only $825 \mu s$ are required for the application of the modified spiral algorithm, including the divisions for calculating the centroids. The division step is realized analogous to Sec. 3.4.2. Special care has to be taken when comparing the values with [16] because the time required for the readout of the camera and identification of the spots is neglected in this publication.

Tab. 3.3 shows the device utilization that is necessary for the detection of the spots (CCL), calculation of the centroids and segmentation based on the spiral algorithm. During the underlying research, the centroid position is calculated with an accuracy of four fractional digits during the division step.

Table 3.6: FPGA logic utilization with $n = 256$, $m = 1$, image width/height 224 pixels (Xilinx Kintex-7 XC7K160T-2FBG676 FPGA) when applying the implementing the modified spiral algorithm [71]

Logic Utilization	Used	Available	Util.
# Slice Registers	4186	202800	2%
# Slice LUTs	5304	101400	5%
# fully used LUT-FF pairs	1801	7689	23%
# Block RAM/FIFO	6	325	1%
# DSP48E1s	24	600	1%

The values of the slice registers, slice LUTs, DSP48E1s and BRAMs in Tab. 3.3 are to be understood without the module for receiving the camera data over the CameraLink interface, deserialization by using the internal SERDES of the FPGA, VGA implementation and PCIe interface. The minimum clock period is 9.511 ns which correlates with a maximum frequency of 105.142 MHz . The desired clocking frequency of 80 MHz has been successfully achieved without further optimization. It is likely that the maximum frequency may be increased further by using optimization

techniques offered by xilinx synthesis technology (XST) such as register duplication. So far, the limiting factor itself is the implementation of the CCL, having relative large routing latencies partly because of the implementation of the stack with FFs.

As platform for the FPGA implementation of the modified spiral algorithm, the own-developed Xilinx Kintex-7 board [69] (shown in Fig. 3.27) has been employed. To this end, the TE0741 micro module of Trenz Electronics [109] based on the Xilinx Kintex-7 XC7K160T-2CF FPGA is used.

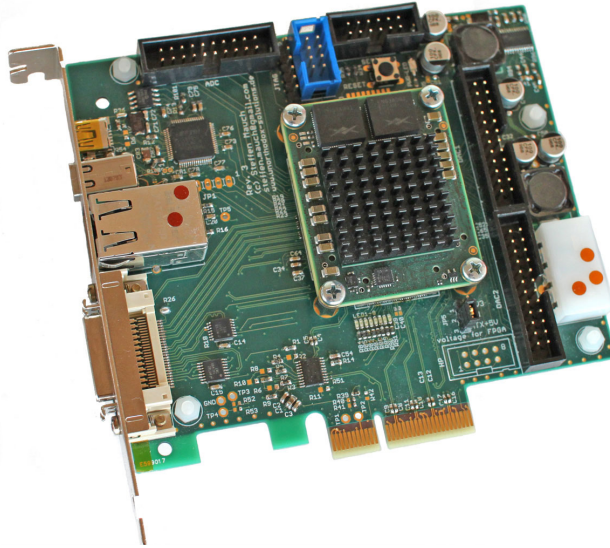


Figure 3.27: Own-developed PCIe board [71], based on the TE0741 Xilinx Kintex-7 FPGA module from Trenz Electronics [109]

As SHWFS, the same camera as previously in Sec. 3.4.2, i.e. a HASOTM3 Fast from Imagine Optics, is used. The same settings are chosen (dual-tap mode with a transmitting frequency of 40 MHz). The ordered and assigned spots can be transmitted to a host by using the integrated PCIe 2.0 x4⁹ interface so as to record and compare the results with other SHWFSs for instance. In addition, it is possible to stream pre-defined images via PCIe to the FPGA for testing the implementation directly. Thus, the implementation may be checked and validated with the MATLAB implementation and the associated results. What is more, the results may be compared with the original spiral algorithm without the need of implementing the original algorithm with VHDL.

The spots are displayed directly via the VGA output enabling a quick check of a correct evaluation of the SHWFS, e.g. in cases when the board is used stand-alone or solely for demonstration purposes.

⁹ PCI Express (Peripheral Component Interconnect Express) is a high-speed serial computer expansion bus standard. PCIe 2.0 x4 has the capability to transfer 2000 MBytes/s.

3.4.4 Real-time adaptive thresholding

The basic principle of the FPGA implementation for the adaptive thresholding is shown in Fig. 3.28. In the case of using the HASOTM3 Fast, see Appendix A.9, the clocking frequency of the sequential pixel stream is 80 MHz.

Equ. (3.23) is identical to Fig. 3.28 but the calculation has been split into two clock cycles for optimizing the achievable speed. This allows reducing the longest combinatorial path and use two clock cycles instead of one, to yield a higher clocking rate. In this case, doubling the frequency to 160 MHz is possible. Therefore the calculation can be split into two clock cycles. Some bit widths are known to be less than specified in Fig. 3.28 because e.g. multiplying two identical numbers will always be positive. Thus the bit width can be decreased by one bit. These optimizations are incorporated in the VHDL implementation of the adaptive thresholding.

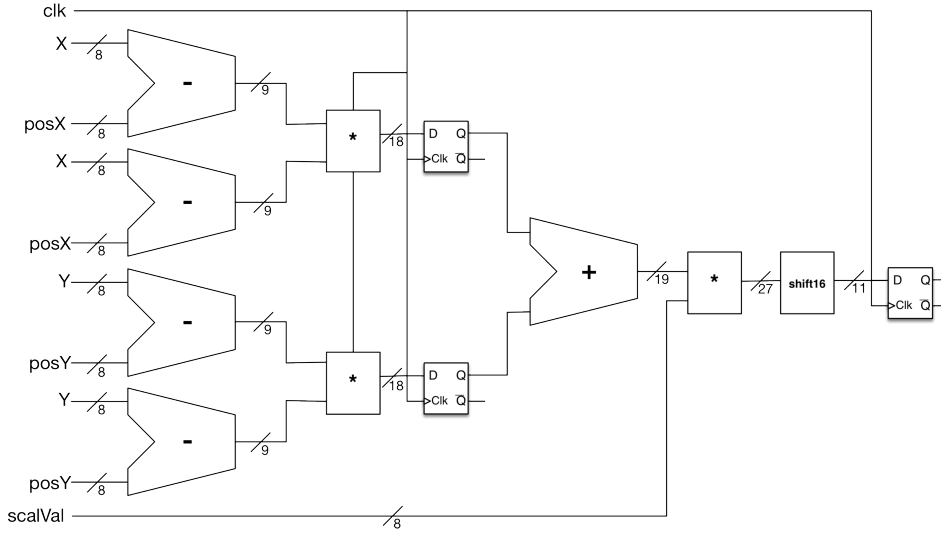


Figure 3.28: Abstract FPGA pipeline implementation of the adaptive thresholding

To unburden the maximum combinatorial path in the CCL component, some experiments have shown that it is beneficial to move the subtraction between the value from Equ. (3.28) and the global threshold value, right before the last FF. By this change, the critical path of the CCL implementation has more timing margin and thus is guaranteed to operate at a higher clocking frequency. To compensate the one clock latency of the computation regarding the 80 MHz clock, the pixel data is delayed by one clock cycle which requires one additional register for storing the pixel value; this additional register is negligible.

By using the presented approach, the maximum clocking frequency of the complete SHWFS evaluation has not been deteriorated because the longest combinatorial path remains the same.

3.4.5 Reordering of centroids by similarity

The proposed method for reordering the centroids by similarity has been implemented in the FPGA as well. In view of the algorithm description in Sec. 3.3.6, nine comparisons between the calibration matrix and the ordered centroids have to be performed to determine the most likely shifting. Instead of a conventional central processing unit (CPU), the FPGA is perfectly suited to perform these comparisons in parallel to reduce the processing time to a minimum. Additionally, not only the nine comparisons can be parallelized, even each individual comparison can be parallelized if the corresponding data is directly available.

The calibration matrix is usually determined during the recording of the AIF (see Sec. 4.2.3). For reordering of the centroids by similarity it is not necessary to perform the ‘AND’ operation with the values of the centroids. Instead, per entry of the centroids matrix, one bit is sufficient to store. This bit represents whether an entry is unequal zero or not. Thus, the corresponding centroid matrix can be converted into a binary matrix. The result is that the calibration matrix only required $14 \cdot 14$ bits in the case of the HASOTM3 Fast SHWFS. The calibration matrix can be written via the PCIe interface. The reordering can be activated as desired via Linux Control and Measurement Device Interface (comedi) over PCIe. The calibration matrix inside the FPGA is stored as a simple register, thus, is accessible in parallel.

During the assignment step of the centroids to the lenslets, the actual centroid matrix (without repositioning) can be converted in parallel into a binary matrix and stored as a register.

The comparison of the two binary matrices is implemented such that fourteen bits are compared within one clock cycle per individual comparison. The sum of the resulting ones is determined within one clock cycle afterwards. To reduce the critical path, the number of ones are summed in an extra clock cycle such that $2 \cdot 14$ clock cycles are required for the complete comparison of one possible shift. For each of the nine possibilities, the module is instantiated in parallel such that, in a subsequent step, the maximum of the nine sums can be determined, thus, the most likely shifting is determined.

Afterwards, the determined shifting value is used to write the centroids into the first in first out (FIFO) such that the centroids information can be written via direct memory access (DMA) into the main-memory of the real-time computer. Additionally, the value of the determined shifting is stored such that the real-time computer can evaluate whether a shifting has been carried out or not.

4 Robust Optimal Control

I highly recommend inviting the worst-case scenario into your life.

PORTIA DE ROSSI

\mathcal{H}_∞ techniques are specific methods for synthesizing controller and filter which have certain defined characteristics concerning robustness for worst-case scenarios. An \mathcal{H}_∞ controller may achieve stabilization of the closed-loop with guaranteed performance even under occurrence of norm-bounded uncertainties. \mathcal{H}_∞ methods are in general formulated as mathematical optimization problems where the resulting controller is the solution that minimizes an \mathcal{H}_∞ norm.

The advantage compared to classical control techniques is based upon the ability to easily handle multiple input multiple output (MIMO) systems with strong cross-couplings between the different channels. For such systems, controllers may be obtained very efficiently within the \mathcal{H}_∞ framework. When the system is a MIMO system, not only designing a controller for one channel is required, instead all channels, i.e. input/output combinations have to be considered simultaneously.

\mathcal{H}_∞ methods require more sophisticated mathematical understanding in order to be successfully applied to a given problem. Furthermore, the system to be controlled has to be reasonably well-known. One has to keep in mind that in practical designs the resulting controller will only be optimal with respect to chosen cost-/weighting functions. Therefore, when speaking about robustness, a model of the physical system as well as a model of the uncertainty or at least a rough approximation of the variations, is always necessary. Otherwise, the design problem may be infeasible or lead to controllers which are highly conservative.

Non-linear constraints, e.g. saturation or hysteresis, cannot be handled well in the standard \mathcal{H}_∞ framework. Primarily \mathcal{H}_∞ control is suitable for linear time-invariant (LTI) systems, but it can be extended to non-linear systems. For the application of \mathcal{H}_∞ in the context of stabilizing DMs, the LTI framework is used with some link to non-linear amendments, e.g. the general small gain theorem where the latter can be extended to the non-linear case as well.

In the 1970s robust control combined the classical methods of the 1940 till 1950s with more sophisticated modern control theory that came up between 1960 and 1970. During the time 1976 till 1981, G. Zames was the first who formulated a basic feedback problem as an optimization problem based on an operator norm, in particular, the \mathcal{H}_∞ norm. Thus, Zames introduced the theory of \mathcal{H}_∞ control already in that time. In 1981, J. C. Doyle and G. Stein showed that internal stability may be preserved

under special perturbations which led to the robust stability design problem [23].

1982 J. C. Doyle argued that uncertainty may often be effectively described in terms of norm-bounded disturbances [24, 81]. Doyle created a truly powerful tool which is known as the structured singular value (SSV) μ by means of which one may test the robust stability in the \mathcal{H}_∞ framework. Additionally, J. C. Doyle presented in 1984 the first solution to a general \mathcal{H}_∞ optimal control problem relying on state-space methods. From that time, \mathcal{H}_∞ optimal control developed rapidly.

In contrast, however, even today PI(D)-based controller designs are mainly used in industry because a lot of practitioners tune the parameter of the controller manually which is, in general, not practicable for MIMO systems of higher order, controlled by an \mathcal{H}_∞ controller. Additionally, the order of the typical \mathcal{H}_∞ controller is the same as of the plant and its weighting filters, thus, the order can be very huge. Therefore, such an \mathcal{H}_∞ controller may require a lot of processing power and may be infeasible for smaller, less powerful systems.

Of course, when the structure of a PI(D) controller is given as a state-space model, the parameter of such a controller can be tuned by application \mathcal{H}_∞ methods. Thus, the PI(D) controller will then be optimal in the sense of \mathcal{H}_∞ . Nevertheless, obtaining a general solution turns out very difficult or even impossible.

In the following sections a rough overview regarding the fundamentals of \mathcal{H}_∞ methods are given. Afterwards, the idea of non-smooth μ -synthesis is introduced which allows to calculate a suboptimal \mathcal{H}_∞ controller for a predefined controller structure.

4.1 \mathcal{H}_∞ Framework

In the following, the nomenclature is loosely based on [100] and references therein [25, 102, 125]. For a more elaborate introduction to the \mathcal{H}_∞ framework the textbooks [124, 125, 102] are recommended.

4.1.1 \mathcal{H}_∞ preliminaries

When dealing with robust control theory for LTI systems, typically, the system is formulated in state-space description. The state-space form of an LTI system reads

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t), & x(0) &= x_0 \\ y(t) &= Cx(t) + Du(t),\end{aligned}\tag{4.1}$$

where the state $x(t) \in \mathbb{R}^n$, output $y(t) \in \mathbb{R}^p$ and input $u(t) \in \mathbb{R}^q$, or in short form notation just $\{A, B, C, D\}$ where the matrices A , B , C and D all have appropriate dimensions.

The input signals u are assumed piecewise continuous functions of time on $t \in [0, \infty)$. For brevity, instead of $x(\cdot)$ often x is written for the signal, whereas $x(t)$ denotes its value at time t .

The system response of (4.1) to an arbitrary input $u(\cdot)$ for $t \geq 0$ can be computed by

$$y(t) = Ce^{At}x_0 + \int_0^t Ce^{A(t-\tau)}Bu(\tau)d\tau + Du(t).\tag{4.2}$$

With the Laplace-transform

$$X(s) = \mathcal{L}\{x(t)\} := \int_0^\infty x(t)e^{-st}dt,\tag{4.3}$$

and likewise the input/output behavior in Laplace domain may be written as

$$Y(s) = \underbrace{\left(C(sI - A)^{-1}B + D\right)}_{G(s)}U(s).\tag{4.4}$$

Equ. (4.4) corresponds to Equ. (4.2) when the initial state $x_0 = 0$. The matrix-valued function of s , $G(s)$, is called transfer function or transfer matrix; clearly revealing its dimensions

$$G(s) = \begin{pmatrix} G_{11}(s) & \dots & P_{1q}(s) \\ \vdots & & \vdots \\ G_{p1}(s) & \dots & P_{pq}(s) \end{pmatrix},\tag{4.5}$$

where any element $G_{ij}(s) = \frac{n_{ij}(s)}{d_{ij}(s)}$ is real-rational, i.e. n_{ij} and d_{ij} are polynomials in s with real coefficients.

Transfer function G is called proper if $\lim_{s \rightarrow \infty} G(s) = \text{const.}$, i.e. $\deg(n_{ij}) \leq \deg(d_{ij}) \forall i, j$. G is called strictly proper if $\lim_{s \rightarrow \infty} G(s) = 0$, i.e. $\deg(n_{ij}) < \deg(d_{ij}) \forall i, j$.

The notation

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right] := G(s)$$

will be used for the mapping from $u \rightarrow y$ as defined via the differential equation with initial condition $x_0 = 0$ as well as for the corresponding transfer matrix $G(s)$.

LTI systems may thus be either represented in the state-space (called time domain) or with a real-rational proper transfer function $G(s)$ (called frequency domain). The fundamental bridge between state-space and frequency domain is established with the realization theory. The step from state-space to frequency domain requires only the calculation of the transfer matrix, see Equ. (4.4). Conversely, given a system in terms of $H(s)$ which is a function of s whose elements are real-rational proper functions, then there always exists matrices A_H , B_H , C_H and D_H such that

$$H(s) = C_H(sI - A_H)^{-1}B_H + D_H. \quad (4.6)$$

The state-space representation of the transfer function $H(s)$ is called a realization. The realization is not unique. The size of the matrix A_H may vary. If matrix A_H is of minimal size, then the realization is called minimal. A simple, sufficient and necessary test for minimality is to verify that (A_H, B_H) is controllable and (C_H, A_H) is observable.

Transfer function $H(s)$ is called stable whenever $H(s)$ is proper and has only poles in the left open complex half-plane. For the set of real-rational, proper and stable transfer functions of dimension $k \times l$, the notion $\mathcal{RH}_\infty^{k \times l}$ is used. A state-space system, Equ. (4.1), is stable whenever all eigenvalues of A are in the open left half-plane \mathbb{C}^- . These stability concepts are closely related. Given a vector-valued signal $u(\cdot)$, signal $u(\cdot)$ is bounded if

$$\|u\|_\infty := \sup_{t \geq 0} \max_i |u_i(t)| \quad (4.7)$$

is finite. Norm $\|u\|_\infty$ is a function norm, called L_∞ -norm.

A system as given in Equ. (4.1) is bounded-input bounded-output (BIBO) stable if an arbitrary bounded input $u(\cdot)$ is mapped onto an output that is bounded as well. Thus $\|u\|_\infty < \infty$ implies $\|y\|_\infty < \infty$. The BIBO stability for LTI systems is equivalent to the stability of the corresponding transfer function. Additionally, another important characteristic is the maximum amplification of signals through a system. When $\|u\|_\infty$ and $\|y\|_\infty$ is taken as a measure for the input and output then the amplification for $u(\cdot)$ is given by the ratio $\frac{\|y\|_\infty}{\|u\|_\infty}$. The worst possible input signal amplification is achieved for the largest quotient when varying $u(\cdot)$ over all bounded

signals [124], resulting in

$$\gamma_{\text{peak}} = \sup_{0 < \|u\|_\infty < \infty} \frac{\|y\|_\infty}{\|u\|_\infty}. \quad (4.8)$$

The value γ_{peak} is the so-called peak-to-peak gain of the system given in Equ. (4.1). Based on the definition of γ_{peak} in Equ. (4.8), $\|y\|_\infty$ can be written by using γ_{peak} and the $\|\cdot\|_\infty$ norm of the input signal $u(\cdot)$, thus

$$\|y\|_\infty \leq \gamma_{\text{peak}} \|u\|_\infty. \quad (4.9)$$

For more information about induced system gains [125, Chapter 4.5] is recommended. In contrast, when using $\|\cdot\|_2$ the energy of a signal can be calculated as follows

$$\|y\|_2 = \sqrt{\int_0^\infty \|x(t)\|^2 dt}. \quad (4.10)$$

Bearing in mind that a signal may have a large energy while having only a small peak, and vice-versa. Therefore, in robust control the $\|\cdot\|_\infty$ norm is considered, focusing at the worst case scenario in a system.

For LTI systems, the question whether a system maps any signal of finite energy onto a signal of finite energy, that is

$$\|u\|_2 \leq \infty \implies \|y\|_\infty \leq \infty \quad (4.11)$$

is equivalent to the BIBO stability property. In this simple form, this property is only valid for LTI systems.

The qualitative property of stability does not depend on the selected measure for the size of the signals, but the system gain itself is dependent on the chosen norm. Defining the energy gain analogously to the previously defined peak-to-peak norm, the energy gain is

$$\gamma_{\text{energy}} = \sup_{0 < \|u\|_2 < \infty} \frac{\|y\|_2}{\|u\|_2}. \quad (4.12)$$

The energy gain of the system may be related to the transfer function of the system. It can be shown that γ_{energy} is equal to the maximal value of

$$\sigma_{\max}(G(j\omega)) = \|G(j\omega)\|_2 \quad (4.13)$$

when maximizing over the frequency $\omega \in \mathbb{R}$ [125, 102]. Introducing

$$\|G\|_\infty := \sup_{\omega \in \mathbb{R}} \sigma_{\max}(G(j\omega)) = \sup_{\omega \in \mathbb{R}} \|G(j\omega)\|_2, \quad (4.14)$$

a norm on the vector space of all real-rational proper and stable transfer functions

$\mathcal{RH}_\infty^{k \times l}$ is defined, called the \mathcal{H}_∞ -norm. Therefore, the energy of a stable LTI system is equivalent to its \mathcal{H}_∞ -norm of the corresponding transfer function

$$\gamma_{\text{energy}} = \|G\|_\infty. \quad (4.15)$$

Linear Fractional Transformation

The linear fractional transformation (LFT), also known as Möbius transformation, is generally a mapping $F : \mathbb{C} \rightarrow \mathbb{C}$ of the following form

$$F(s) = \frac{a + bs}{c + ds} \quad (4.16)$$

where $a, b, c, d \in \mathbb{C}$. If $c \neq 0$, $F(s)$ can also be written as

$$F(s) = \alpha + \beta s(1 - \gamma s)^{-1} \quad (4.17)$$

for $\alpha, \beta, \gamma \in \mathbb{C}$. So far, the LFT is defined for the case that $a, b, c, d \in \mathbb{C}$ which is the scalar case. Nevertheless, the LFT can be generalized to the matrix case.

Suppose P and Δ_l are given transfer matrices, the lower LFT $\mathcal{F}_\ell(P, \Delta_l)$ of P and K is defined as follows. Partition

$$P = \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix} \quad (4.18)$$

such that $P_{22}\Delta_l$ is square and check if the rational matrix $I - P_{22}\Delta_l$ has a rational inverse. Then define [125]

$$\mathcal{F}_\ell(P, \Delta_l) := P_{11} + P_{12}\Delta_l(I - P_{22}\Delta_l)^{-1}P_{21}. \quad (4.19)$$

The upper LFT $\mathcal{F}_u(\Delta_u, P)$ of the rational matrices Δ_u and P is defined in the following way. Again partition

$$P = \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix} \quad (4.20)$$

such that $P_{11}\Delta_u$ is square. If the rational matrix $I - P_{11}\Delta_u$ has a rational inverse then define

$$\mathcal{F}_u(P, \Delta_u) := P_{22} + P_{21}\Delta_u(I - P_{11}\Delta_u)^{-1}P_{12}. \quad (4.21)$$

The naming convention of $\mathcal{F}_\ell(P, \Delta_l)$ and $\mathcal{F}_u(P, \Delta_u)$ is evident reckoning the following

illustrations (Fig. 4.1a and 4.1b). Fig. 4.1a represents Equ. (4.22)

$$\begin{pmatrix} z_1 \\ y_1 \end{pmatrix} = P \begin{pmatrix} w_1 \\ w_1 \end{pmatrix} = \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix} \begin{pmatrix} w_1 \\ u_1 \end{pmatrix} \quad (4.22)$$

$$u_1 = \Delta_l y_1$$

whereas

$$\begin{pmatrix} y_2 \\ z_2 \end{pmatrix} = P \begin{pmatrix} w_2 \\ w_2 \end{pmatrix} = \begin{pmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{pmatrix} \begin{pmatrix} u_2 \\ w_2 \end{pmatrix} \quad (4.23)$$

$$u_2 = \Delta_u y_2$$

represents Fig. 4.1b

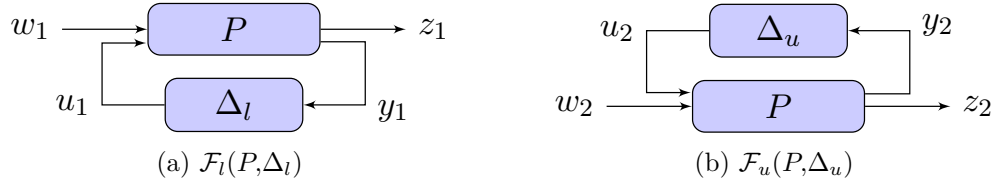


Figure 4.1: Graphical representation of the upper- and lower LFT

LFTs are especially useful when studying the influence of perturbations on a given system. The LFT is stating a mapping $w_1 \mapsto z_1$ and $w_2 \mapsto z_2$ whereas $\Delta_{l/u}$ is either a system model uncertainty or the controller itself.

Robustness & Performance

The purpose of robust control can be stated as the circumstance that the closed-loop performance should remain acceptable when perturbations occur, e.g. in the plant. Let $P_\Delta(s)$ denote the set of all perturbed plants and $P_0(s)$ the nominal plant. The controller $K(s)$ should stabilize the closed-loop and guarantee to preserve the performance in a specified vicinity around the nominal closed-loop of P_0 and $K(s)$.

In total, four different kinds of specifications [100, 125] are used commonly:

- nominal stability
For the nominal plant P_0 the closed-loop is stable.
- nominal performance
For the nominal plant P_0 the closed-loop specification holds.
- robust stability
For the plants given in the set P_Δ the closed-loop is stable.
- robust performance
For the plants given in the set P_Δ the closed-loop specification holds.

The difference between stability and performance is the fact that when speaking of performance, parameters such as e.g. settling time, overshoot etc. are given, whose imposed constraints need to be met.

Small Gain Theorem

When analyzing non-linear systems, the formalism of input-output stability is an important tool in studying the stability of interconnected systems [49]. Due to the fact that the gain of a system directly relates to whether the norm of a signal increases or decreases as the signal traverse through the system, the formalism of input-output stability is very useful. The so called small-gain theorem is a sufficient condition for finite-gain \mathcal{L} stability of the interconnection of two systems. The theorem was proved by George Zames in 1966. The small-gain theorem can be seen as a generalization of the Nyquist criterion to non-linear time-varying MIMO systems [121, 120].

For the application of robust control for LTI systems in the context of \mathcal{H}_∞ control, the following slightly modified version of the small gain theorem is commonly used.

Theorem 1 (Small Gain Theorem)

Suppose $P \in \mathcal{RH}_\infty$ and let $\gamma > 0$. Then the interconnected system, as shown in Fig. 4.2, is well-posed and internally stable for all $\Delta \in \mathcal{RH}_\infty$ with

1. $\|\Delta\|_\infty \leq 1/\gamma$ if and only if $\|P\|_\infty < \gamma$
2. $\|\Delta\|_\infty < 1/\gamma$ if and only if $\|P\|_\infty \leq \gamma$.

One potential interpretation is the following: If the \mathcal{H}_∞ -norm of P decreases, the radius of the admissible uncertainty increases and vice-versa.

If the small gain theorem is formulated for two stable transfer functions S_1 and S_2 then the interconnection of the closed-loop system is input-output stable whenever

$$\|S_1\|_\infty \cdot \|S_2\|_\infty \leq 1 \quad (4.24)$$

holds. In the case that S_1 and S_2 correspond to LTI systems, only $\|S_1 S_2\|_\infty \leq 1$ has to be satisfied instead of checking Equ. (4.24).

In general, the norm of Theorem 1 may either be the infinity norm or any other induced norm; the result will remain the same.

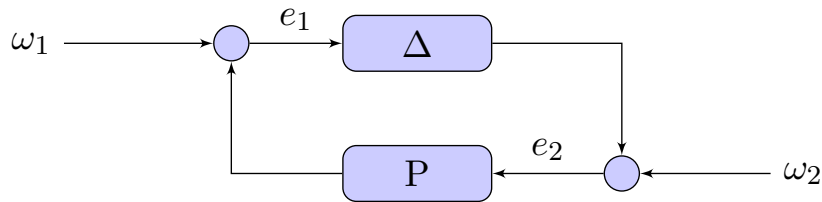


Figure 4.2: Block diagram of the interconnection of Δ and P in the context of the small gain theorem

4.1.2 Handling uncertainty

When identifying or modeling a system, generally, a single description for a plant is unrealistic. Instead, a set of valid plants, denoted G_Δ , is the outcome of a typical identification process. On the one hand, this is due the presence of measurement noise, and at the other hand, due to parameter variations and lack of detailed characterizations or model complexity.

For incorporating G_Δ in the controller synthesis process, a mathematical description of this set is required. Additionally, only physically meaningful model errors should be involved. Otherwise, the synthesis will try to include dispensable uncertainties likewise which is generally not desired.

The main distinction regarding the classification of uncertainty is given by the differentiation between unstructured and structured uncertainties. A parameter uncertainty is structured, e.g. by restricting the parameter range to some given bounded interval. Bearing in mind a correct model for the uncertainty, yields a better understanding for matching of the uncertainty. Nevertheless, a detailed description of uncertainties complicates the synthesis process of the controller due to the increasing complexity. This contradicts an elaborate uncertainty description. A good tradeoff between elaborate description and manageable model dimension is a key for achieving good synthesis results for the controller and required computational power during the synthesis process.

Parametric Uncertainties

In cases when a nominal parameter is able to take different values, but does not change with time (parameter is time-invariant), then the following scheme may be applied for modeling the parametric uncertainty.

As an example, a scalar second order differential equation

$$T^2\ddot{y}(t) + 2dT\dot{y}(t) + y(t) = u(t) \quad (4.25)$$

is analyzed where the damping constant is given by

$$d_1 \leq d \leq d_2$$

for some real number $0 \leq d_1 \leq d_2$. Furthermore, $y(\cdot)$ denotes the output, $u(\cdot)$ the input and T a fixed parameter. The nominal value d_0 is chosen as $d_0 := \frac{d_1+d_2}{2}$. As scaled error variable

$$\Delta = \frac{d - d_0}{d_2 - d_0} \quad (4.26)$$

is introduced.

Due to this choice, the property

$$d = d_0 + W\Delta \quad \text{with} \quad W = d_2 - d_0 \quad (4.27)$$

holds. The class of uncertainties can now be defined as follows

$$\Delta := \{\Delta \in \mathbb{R} \mid -1 \leq \Delta \leq 1\}. \quad (4.28)$$

The corresponding uncertain system in the frequency-domain is given as

$$G_\Delta(s) = \frac{1}{T^2 s^2 + 2(d_0 + W\Delta)Ts + 1}. \quad (4.29)$$

By expression $G_\Delta(s)$ a whole set of transfer functions is given, parametrized through Δ by varying the set Δ . For this step, the original parameter $d \in (d_1, d_2)$ has been transformed to the new parameter $\Delta \in (-1, 1)$ by considering the nominal value d_0 and the corresponding weight W .

Bear in mind that Equ. (4.29) is only valid when the uncertainty is time-invariant.

Dynamic uncertainties

Dynamic uncertainties are incurred when, for example, a model is identified based on measurements resulting from injecting sinusoidal signals. Typically, the frequency response measurements are flawed with uncertainty. Having multiple measurements for a given frequency ω , a set of complex numbers (denoted by $\mathcal{H}(\omega)$) is determined instead of a single complex number. Of course, methods such as averaging may be applied to the set $\mathcal{H}(\omega)$. Nevertheless, it is likely that this results in an inconsistent frequency response. The problem appears due to the finite number of measurements and because inaccuracies such as noise do not necessarily satisfy the properties of white noise. Thus, these inaccuracies which are included in the measurement do not have an arithmetic mean of zero.

Notwithstanding, it is self-evident that any proper and stable transfer function $H(s)$ which satisfies

$$H(j\omega) \in \mathcal{H}(\omega)$$

may be considered a convenient model for the underlying plant. But still, the main problem remains the same since no appropriate description for the set $\mathcal{H}(\omega)$ is prevalent. Therefore, even when multiple valid descriptions are available, these are not suitable for development of any further theory. Multiple descriptions do not allow an overall consistent description which is required.

Similar to the approach when handling parametric uncertainties, the idea is to cover $\mathcal{H}(\omega)$ with a real rational proper transfer matrix $G(s)$ and an uncertainty such that

$$\mathcal{H}(\omega) \in G(j\omega) + W(j\omega)\Delta_c \quad \text{for all} \quad \omega \in \mathbb{R} \cup \{\infty\}.$$

Δ_c is the open unit disk around point 0 with $\Delta_c := \{\Delta_c \in \mathbb{C} \mid |\Delta_c| < 1\}$ and $W(s)$ a real rational weighting function. Δ_c does not necessarily have to be a unit disk, instead e.g. a polytope containing point the 0 is possible as-well. Additionally, the set of values Δ_c may be written as $\Delta_c(\omega)$ incorporating more precisely the phase information of the uncertainty.

$G(j\omega)$ admits the interpretation as a nominal system. The deviation from $G(j\omega)$ is determined by the circle $W(j\omega)\Delta_c$ whereas the radius of $|W(j\omega)|$ varies over frequency. In general, W is a high-pass filter because identified models lack typically accuracy at higher frequencies due to different reasons such as limited bandwidth and lower signal amplitudes in that range.

The actual set of uncertainties is defined as

$$\Delta := \{\Delta(s) \in \mathcal{RH}_\infty \mid \Delta(j\omega) \in \Delta_c \text{ for all } \omega \in \mathbb{R} \cup \{\infty\}\}. \quad (4.30)$$

This set is often called the open unit ball in \mathcal{RH}_∞ , that is

$$\Delta := \{\Delta(s) \in \mathcal{RH}_\infty \mid \|\Delta\|_\infty < 1\}. \quad (4.31)$$

4.1.3 Structured singular value (μ)

Parametric and dynamic uncertainties, which have been introduced before, can be pulled out of a description by application of the LFT. One of many possibilities is defining the structure as following

$$\Delta_c = \left\{ \text{diag}(\delta_1 I_{n_1}, \dots, \delta_{r_s} I_{n_{r_s}}, \Delta_1, \dots, \Delta_{n_f}) \mid \delta_i \in \mathbb{C}, \Delta_k \in \mathcal{RH}_\infty^{p_k \times q_k}; \right. \\ \left. |\delta_i| < 1 \text{ for } i = 1, \dots, r_s, \|\Delta_k\|_\infty < 1 \text{ for } k = 1, \dots, n_f \right\} \quad (4.32)$$

where δ_i is complex block for $n_i = 1$. If $n_i \neq 1$, it is a complex repeated block. Δ_k is a full unstructured dynamic block. Of course, the repetitions (dimensions of the identities) and the dimension of the full blocks can all be diverse. The set of all these complex matrices is denoted with Δ_c .

The actual set of uncertainties Δ is the set of all stable and real rational proper Δ which frequency response accepts values in Δ_c

$$\Delta := \left\{ \Delta(s) \in \mathcal{RH}_\infty \mid \Delta(j\omega) \in \Delta_c \quad \forall \quad \omega \in \mathbb{R} \cup \{\infty\} \right\}$$

Due to the definition of the structure as in Equ. (4.32), one can write $\|\Delta_c\|_\infty < 1$. Thus the set $r\Delta_c$ consists of all complex matrices Δ_c and is bounded by $\|\Delta_c\|_\infty < r$.

After pulling out the uncertainties and using as description the set Δ , the system can be drawn as shown in Fig. 4.3, consisting of Δ and the resulting plant $P(s)$. The

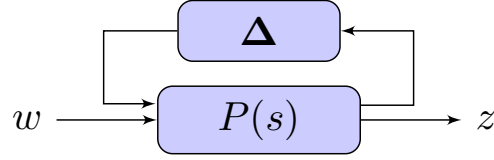


Figure 4.3: Interconnection of $P(s)$ and Δ for calculation of the structured singular value μ

dependence between the disturbance w and the output z is given by the relationship

$$z = \mathcal{F}_u(P, \Delta) = \left[P_{22} + P_{21} \Delta (I - P_{11} \Delta)^{-1} P_{12} \right] w. \quad (4.33)$$

The challenge is to judge whether the interconnection of $P(s)$ and Δ_c is stable for all elements of the given set Δ_c . Therefore the expression $(I - P_{11}(j\omega)\Delta_c)^{-1}$ must exist and be stable (bounded) for each element for all $\Delta_c \in \Delta_c$. Testing robust stability against structured uncertainties may be reduced to check robust non-singularity against structured matrices.

The following tests are straight-forward but come with the disadvantage that in general these characteristics are not sufficient:

1. $\|P_{11}\|_\infty \leq 1$ implies stability due to the small gain theorem but not conversely (as e.g. block diagonal structure of uncertainties are ignored). Therefore, this criterion can be arbitrarily conservative.
2. Test each δ_i/Δ_j individually (e.g. assuming that no uncertainties in other channels are present). However, this test can be arbitrarily optimistic because the interaction between blocks in the uncertainty matrix is completely ignored.

Due to the arbitrarily conservative or optimistic tests, other methods to judge the stability are required to be applied.

r^* denotes the largest r for which $I - P_{11}\Delta_c$ is non-singular for all $\Delta_c \in \Delta_c$. Then, the structured singular value μ of the complex-valued matrix P_{11} with respect to the set Δ_c is defined as following

$$\mu_{\Delta_c}(P_{11}) = \frac{1}{r^*} = \frac{1}{\sup \{r \mid \det(I - P_{11}\Delta_c) \neq 0 \ \forall \ \Delta_c \in r\Delta_c\}}. \quad (4.34)$$

If the denominator $\det(I - P_{11}\Delta_c) \neq 0 \ \forall \ \Delta_c \in \Delta_c$ some elementary properties i.e.

1. $\Delta_c = \mathbb{C}^{p \times q} \Rightarrow \mu_{\Delta_c}(P_{11}) = \bar{\sigma}(P_{11})$
2. $\Delta_c = \{\delta I : \delta \in \mathbb{C}\} \Rightarrow \mu_{\Delta_c}(P_{11}) = \rho(P_{11})$
3. in general, $\mathbb{C} \subset \Delta_c \subset \mathbb{C}^{p \times q}$ so $\rho(P_{11}) \leq \mu_{\Delta_c}(P_{11}) \leq \bar{\sigma}(P_{11})$

are always true whereas $\rho(\cdot)$ denotes the spectral radius. Instead of writing $\mu_{\Delta_c}(\cdot)$, simply $\mu(\cdot)$ or $\mu(\Delta_c, \cdot)$ is used often. Nevertheless, it is important to remember that another choice of Δ_c would lead to another $\mu(\cdot)$ value.

Based on these elementary properties, it is obvious that the singular value itself is not a good bound indeed too. Due to the inequality (see third property), one idea is to find a transformation which does not affect $\mu(P_{11})$ but change $\rho(P_{11})$ and $\bar{\sigma}(P_{11})$ instead.

Therefore the two following sets are defined

$$\mathcal{U} = \{U \in \Delta_c : UU^* = I\} \quad (4.35)$$

$$\begin{aligned} \mathcal{D} = \{ \text{diag}(D_1, \dots, D_K, d_1 I_{m_1}, \dots, d_{L-1} I_{m_l}) : D_k \in \mathbb{C}^{r_k \times r_k}, \\ D_k = D_k^* \geq 0, d_L \in \mathbb{R}, d_L \geq 0 \}. \end{aligned} \quad (4.36)$$

Worthwhile for any $\Delta_c \in \Delta_c$, $U \in \mathcal{U}$ and $D \in \mathcal{D}$ the following properties hold.

1. $U^* \in \mathcal{U}$, $U\Delta_c \in \Delta_c$, $\Delta_c U \in \Delta_c$ consequence of the property of the set Δ_c
2. $\|U\Delta_c\| = \|\Delta_c U\| = \|\Delta_c\|$ since $UU^* = I$
3. $D\Delta_c = \Delta_c D$ consequence of the property of the set \mathcal{D}

The definition of \mathcal{U} and \mathcal{D} is the same as used in Equ. (4.35) and (4.36).

Referring to the application of the structured singular value μ , the following theorem holds:

Theorem 2

For all $U \in \mathcal{U}$ and $D \in \mathcal{D}$

1. $\mu(P) = \mu(UP) = \mu(PU)$
2. $\mu(P) = \mu(DPD^{-1})$.

The proof of theorem 2 is given in the following:

Proof 1 (for Theorem 2)

1. Since for each $U \in \mathcal{U}$

$$\det(I - P\Delta_c) = 0 \iff \det(I - PUU^*\Delta_c) = 0$$

$$\Delta_c \in \Delta_c \iff U^*\Delta_c \in \Delta_c$$

$$\mu(P) = \mu(PU).$$

2. For all $D \in \mathcal{D}$

$$\det(I - P\Delta_c) = \det(I - PD^{-1}\Delta_c D) = \det(I - DPD^{-1}\Delta_c)$$

$$\text{since } \Delta_c \text{ and } D \text{ commute. Therefore } \mu(P) = \mu(DPD^{-1}).$$

It is worth mentioning that determining the value of the structured singular value μ is a convex problem. Thus, the determination of μ for a given problem is feasible easily. By calculating μ for a given system P_{11} , one may test whether the interconnection of Δ_c and P_{11} is stable, but the problem of finding a stabilizing controller is not tackled so far by using the structured singular value μ .

4.1.4 \mathcal{H}_∞ optimal controller design

With the basic theory, introduced in the preceding exposition, the knowledge may now be applied to synthesize an \mathcal{H}_∞ optimal controller.

To this end, suppose the generalized plant P

$$\begin{aligned} \dot{x} &= Ax + B_1w + B_2u \\ z &= C_1x + D_{11}w + D_{12}u \\ y &= C_2x + D_{21}w \end{aligned} \tag{4.37}$$

and the corresponding controller K as

$$\begin{aligned} \dot{x}_K &= A_Kx_K + B_Ky \\ u &= C_Kx_K + D_Ky. \end{aligned} \tag{4.38}$$

Then by application of the LFT, the interconnection of P and K (closed-loop system) is given by $\mathcal{F}_l(P, K)$, thus

$$\begin{aligned} \dot{\zeta} &= \mathcal{A}\zeta + \mathcal{B}w \\ z &= \mathcal{C}\zeta + \mathcal{D}w \end{aligned} \tag{4.39}$$

where \mathcal{A} , \mathcal{B} , \mathcal{C} , \mathcal{D} is given by

$$\left[\begin{array}{cc|c} A + B_2D_KC_2 & B_2C_K & B_1 + B_2D_KD_{21} \\ B_KC_2 & A_K & B_KD_{21} \\ \hline C_1 + D_{12}D_KC_2 & D_{12}C_K & D_{11} + D_{12}D_KD_{21} \end{array} \right]. \tag{4.40}$$

For gathering Equ. 4.40, the assumption is that D_{22} is equal to zero. This is not an restriction since D_{22} can be pushed into the controller K . D_{22} would extend Equ. (4.37) in such a way that the output y is directly influenced by the input u .

Based on the composition of P and K , the goal is now to minimize

$$\|\mathcal{F}(P, K)\|_\infty = \left\| \mathcal{C}(sI - \mathcal{A})^{-1} \mathcal{B} + \mathcal{D} \right\|_\infty \tag{4.41}$$

over all controller fulfilling the property that \mathcal{A} has all eigenvalues in \mathbb{C}^- . Any controller K must render the eigenvalues of \mathcal{A} , $\lambda(\mathcal{A})$ in \mathbb{C}^- and the resulting closed-loop system is stable.

A controller is called γ -suboptimal when

$$\left\| \mathcal{C}(sI - \mathcal{A})^{-1} \mathcal{B} + \mathcal{D} \right\|_\infty < \gamma, \quad \lambda(\mathcal{A}) \in \mathbb{C}^-$$

for some given number $\gamma > 0$. To compute the minimum possible γ , the bisection method can be applied to approximate the value arbitrarily close (see Appendix A.4). Nevertheless, this method can only be applied when already having a suitable con-

troller K .

So far, no restriction prevails as $D_{22} = 0$ can be reached by a simple transformation. A straight forward solution for solving Equ. (4.41) exists when the following properties are fulfilled:

- (i) (A, B_1) has no uncontrollable and (C_1, A) no unobservable mode on $j\omega$
- (ii) (A, B_2) is stabilizable and (C_2, A) is detectable
- (iii) $D_{12}^T \begin{pmatrix} C_1 & D_{12} \end{pmatrix} = \begin{pmatrix} 0 & I \end{pmatrix}$, $\begin{pmatrix} B_1 \\ D_{21} \end{pmatrix} D_{21}^T = \begin{pmatrix} 0 \\ I \end{pmatrix}$
- (iv) $D_{11} = 0$ and $D_K = 0$ (meaning strictly proper controllers) which implies $\mathcal{D} = 0$.

The assumption (i) is of technical reason as together with (ii) it guarantees that the Hamiltonian matrices are indeed solutions. (ii) is necessary and sufficient that P can be internally stabilized by output feedback; (i) guarantees that the internal stability of P is equivalent to the BIBO stability of w to z . (iii) and (iv) simplify the derivation and can be generalized by input-/output transformations see [125, Chapter 16].

The corresponding solution for the suboptimal \mathcal{H}_∞ problem is given by the following theorem.

Theorem 3

Given the afore-stated assumptions, there exists a suboptimal \mathcal{H}_∞ controller if and only if the Algebraic Riccati Equations (AREs) (see Appendix A.2) and [125, Chapter 16.2],

$$\begin{aligned} A^T X + X A + X \left(\frac{1}{\gamma} B_1 B_1^T - B_2 B_2^T \right) X + C_1^T C_1 &= 0 \\ A Y + Y A^T + Y \left(\frac{1}{\gamma} C_1^T C_1 - C_2^T C_2 \right) Y + B_1 B_1^T &= 0 \end{aligned} \quad (4.42)$$

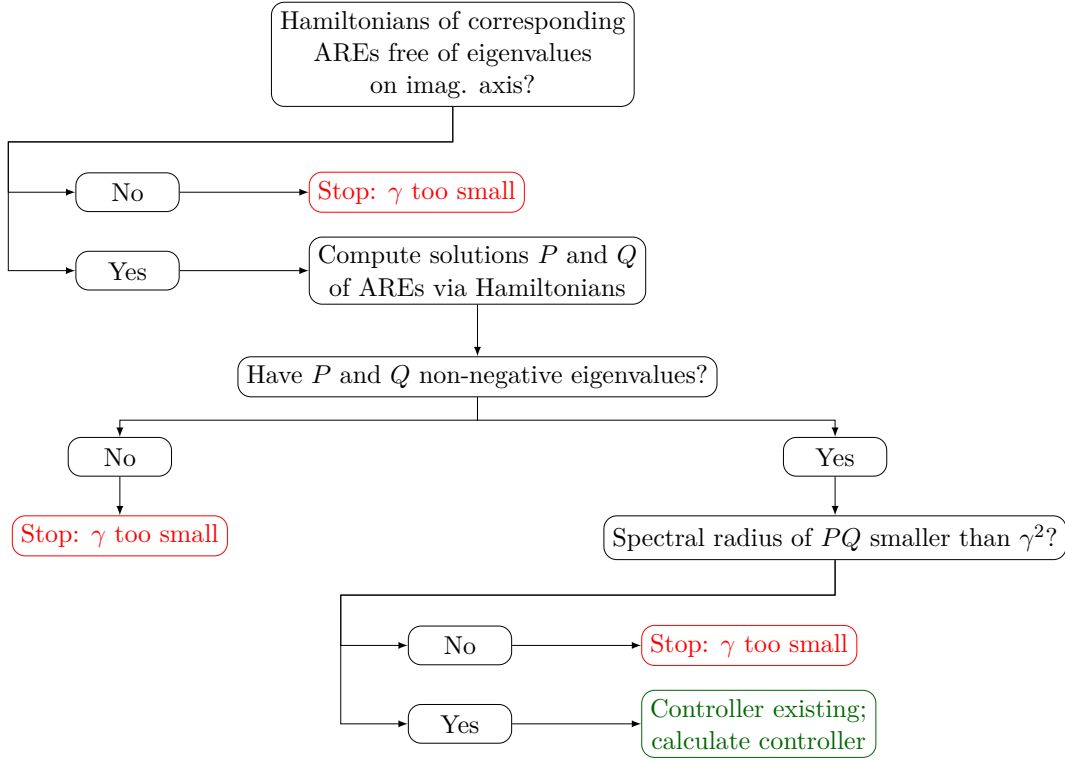
have stabilizing solutions X_- , Y_- and $\rho(X_- Y_-) < \gamma^2$ is satisfied.

ρ denotes the spectral radius. Fig. 4.4 illustrates the procedure for solving the problem with the AREs from Equ. (4.41), given a chosen value for γ .

Generic \mathcal{H}_∞ constraints can also be converted into Algebraic Riccati Inequalities (ARIs) by using the Bounded Real Lemma (see Appendix A.3).

The absolute minimum required properties (e.g. requirements of the ‘hinfsyn’ command of MATLAB) are

1. (A, B_2) is stabilizable and (C_2, A) is detectable
2. D_{12} has full column rank and D_{21} has full row rank
3. $\begin{pmatrix} A - i\omega I & B_2 \\ C_1 & D_{12} \end{pmatrix}$ and $\begin{pmatrix} A - i\omega I & B_1 \\ C_2 & D_{21} \end{pmatrix}$ do not have a rank deficiency for any $\omega \in \mathbb{R}$


 Figure 4.4: Summary for calculating the output feedback \mathcal{H}_∞ controller

The second and third hypotheses can be relaxed by matrix perturbation [100, 99]. Worth mentioning is the fact that the resulting dimension of the controller is equal to the plant itself.

To this end, the \mathcal{H}_∞ problem has not incorporated uncertainties so far. Note that until now just for a specific plant, but not for set of plants, a controller K may be calculated.

In the general case, the transfer function from $W(s)$ to $Z(s) = \begin{pmatrix} Z_o(s) \\ Z_u(s) \end{pmatrix}$ results from

$$Z(s) = \mathcal{F}_\ell(\mathcal{F}_u(P(s), \Delta), K(s))W(s) = T_{zw}(s)W(s) \quad (4.43)$$

with \mathcal{F}_ℓ and \mathcal{F}_u denoting the lower and upper LFTs, resp. Fig. 4.5 visualizes the relationship between uncertainty Δ , plant $P(s)$ and controller $K(s)$.

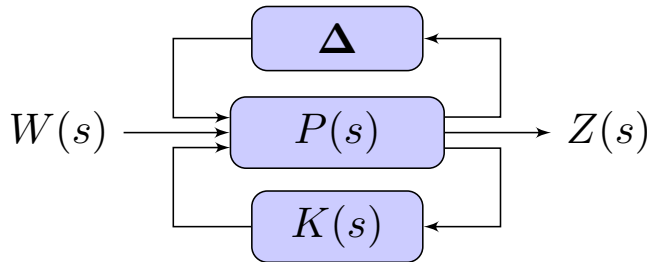


Figure 4.5: Block diagram with uncertainty, plant and controller

Due to the inclusion of uncertainty, the previous result alone is not suitable for solving the \mathcal{H}_∞ problem. As previously discussed, the structured singular value μ may cope with that task. Thus, the problem can be reformulated by minimizing μ of the transfer function $T_{zw}(s)$ and Δ . In other words, minimizing the influence of the uncertainty on the transfer function from the input disturbance w to the output y under consideration of all covered uncertainties of the set Δ .

When $T_{zw}(s)$ is given by pulling out Δ , $\mu(\Delta, T_{zw})$ has to be calculated as follows

$$\mu(\Delta, T_{zw}) = \frac{1}{\min \{\bar{\sigma}(\Delta) : \det(I - T_{zw}\Delta) = 0\}}. \quad (4.44)$$

By application of the small gain theorem, the aspect of stability can be judged while using the structured singular value μ . The interconnection of $T_{zw}(s)$ and Δ is stable when the following equation holds

$$\bar{\sigma}(\Delta)\mu(\Delta, T_{zw}) < 1 \quad \forall s = j\omega. \quad (4.45)$$

Of course this inequality is only sufficient. Thus, if the inequality is not fulfilled it is not for sure that the interconnection is not stable.

Nevertheless, Equ. (4.45) is easy to be checked but does not solve the problem of finding a stabilizing controller with respect to the \mathcal{H}_∞ norm. Therefore, further methods are necessary to obtain a controller K which minimizes the μ value of $T_{zw}(s)$ which incorporates the controller K . These are two problems which are depended of each other to some kind. Each problem itself is a convex problem meaning that minimizing μ while not changing $K(s)$ and minimizing the \mathcal{H}_∞ norm while neglecting Δ is feasibly by e.g. using linear matrix inequality (LMI) optimization techniques.

4.1.5 Controller design via non-smooth μ -synthesis

The structured singular value μ (Sec. 4.1.3) is a very powerful tool for the analysis of robust performance when the controller $K(s)$ is already determined. An obvious straight-forward idea is to synthesize a \mathcal{H}_∞ -controller which minimizes the structured singular value μ .

One of the first attempts in the past has been the so called D/K iteration, which is briefly explained in Appendix A.5. The main idea is based on the combination of classical \mathcal{H}_∞ -synthesis and μ -analysis. The D/K iteration decisively depends on the optimality of both approaches but the iteration may still lead to not satisfactory results since the convergence cannot be guaranteed in general. Additionally, the resulting controller has, at least, the same dimension as the overall plant which can be problematic when having models with large orders.

In the past, several approaches have been introduced to obtain a fixed-order \mathcal{H}_∞ optimal controller while incorporating uncertainty in the controller synthesis. Recently, a new approach has been presented which, at the one hand, combines the \mathcal{H}_∞ -synthesis and μ -analysis into a single problem. On the other hand, the approach

can handle fixed-order controllers at the same time. The approach is called non-smooth μ -synthesis [28, 7, 8, 9, 27, 87, 11]. As the name of the algorithm suggests, the solution is based on non-smooth optimization using Clarke subdifferentials. This approach tries to solve a non convex problem as do the other combined approaches. Several examples have demonstrated the good results which can be achieved with the novel approach (e.g. [87, 66, 93] to name but a few).

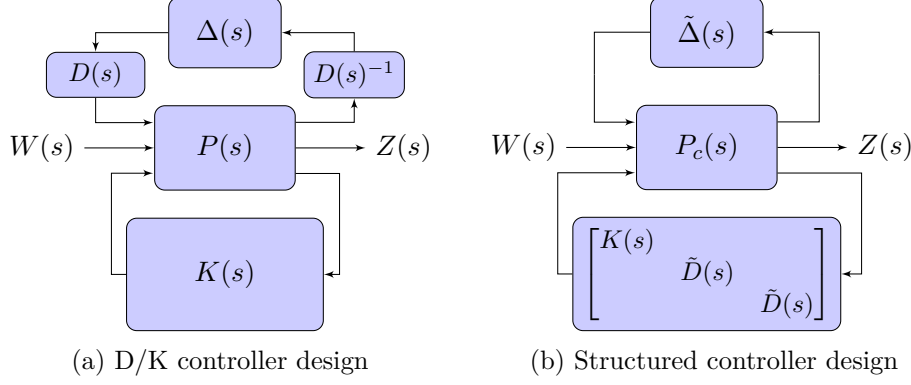


Figure 4.6: Loop equivalence when the D-scaling transformation is applied for structured controller design

Given the traditional approach using the D/K iteration (shown in Fig. 4.6a), the structure and order of $K(s)$ cannot be influenced without changing the plant description. Also instead of $D(s)$, a D_ω is calculated. Thus, some points are calculated during each step of the D/K iteration instead of a realization $D(s)$. Afterwards these individual points are used for identification of a transfer function $D(s)$, see Appendix A.5. Utilizing some loop transformation, Fig. 4.6a can be transformed into Fig. 4.6b whereas $\tilde{D}(s)$ and $K(s)$ could be arbitrary chosen regarding the structure and order of the transfer function. The loop transformation leads to a duplication of the D-scaling; therefore $\tilde{D}(s)$ is appearing twice in Fig. 4.6b. The special arrangement in Fig. 4.6b is named structured controller design. The abstract controller consisting of $\text{diag}(K(s), \tilde{D}(s), \tilde{D}(s))$ has repeated diagonal structure.

Therefore the following equation used for the D/K iteration

$$\bar{\mu} = \inf_{\substack{D \in \mathcal{D} \\ K(s) \text{ proper, stabilizing}}} \|D \mathcal{F}_\ell(P, K) D^{-1}\|_\infty \quad (4.46)$$

can now be written as

$$D(j\omega) \mathcal{F}_\ell(P(j\omega), K(j\omega)) D(j\omega)^{-1} = F_\ell \left(P_c(j\omega), \begin{pmatrix} K(j\omega) & 0 & 0 \\ 0 & \tilde{D}(j\omega) & 0 \\ 0 & 0 & \tilde{D}(j\omega) \end{pmatrix} \right). \quad (4.47)$$

This approach can be extended to multiplier-based and integral quadratic constraint

(IQC) synthesis. The main difficulties are the fact that global solutions are not systematically accessible, in general calculating a solution is NP-hard and thus clearly non-convex. The usual approaches based on LMI and Riccati methods could not be applied as these require in general convex problems. bilinear matrix inequality (BMI) approaches are unreliable and costly. Of course, also the D/K iterations may fail to converge. For many systems, indeed, the D/K iteration is far from optimal but converges to a stabilizing controller while incorporating uncertainty in the synthesis step.

Non-smooth μ -synthesis

It is not constructive to dive into the full details of the non-smooth μ -synthesis approach as this method requires a lot of knowledge concerning non-smooth optimization which itself depends on a good fundamental mathematical background. The main focus of this dissertation is kept more on the application side. Nevertheless, a basic understanding of the non-smooth μ -synthesis is necessary for successful application.

The differentiation between structured and unstructured controller design is fundamental. Defining the controller as in Equ. (4.38), K is called structured if the matrices A_K, B_K, C_K, D_K depend smoothly on a design parameter $\mathbf{p} \in \mathbb{R}^{n_p}$. \mathbf{p} is called the vector of tunable parameters and is different to the state vector x_K . The number of parameters is denoted with n_p . In fact, instead of writing A_K, B_K, C_K, D_K better write $A_K(\mathbf{p}), B_K(\mathbf{p}), C_K(\mathbf{p}), D_K(\mathbf{p})$. Often the notation $K(\mathbf{p})$ is introduced to highlight that the controller is parametrized with \mathbf{p} .

The structured controller synthesis problem without considering uncertainty can be formulated as an optimization problem as following

$$\begin{aligned} & \text{minimize} && \|T_{zw}(P, K(\mathbf{p}))\| \\ & \text{subject to} && K(\mathbf{p}) \text{ closed-loop stabilizing} \\ & && K(\mathbf{p}) \text{ structured, } \mathbf{p} \in \mathbb{R}^{n_p} \end{aligned} \tag{4.48}$$

As previous, $\|\cdot\|$ can be any function norm such as e.g. \mathcal{H}_∞ or \mathcal{H}_2 norm. The counterpart of the structured controller synthesis problem is called unstructured synthesis or referred as black-box controller.

The structured concept can also be applied for incorporating uncertainty as this has already been used in Fig. 4.6b. Later sections and chapter will instead of writing $K(\mathbf{p})$, simply use K (see Fig. 4.6b) as the underlying structure of K is always identical.

Since the late 1990s several researcher have worked on the field of design techniques for structured controllers [12]. Some approaches have been using bilinear matrix inequalities, others local optimization techniques, to name but a few. These techniques have in common that their success have been limited.

Another approach has been the non-smooth optimization [7, 77, 10] In general, a

non-smooth optimization problem has the following form

$$\begin{aligned} & \text{minimize} && f(\mathbf{p}) \\ & \text{subject to} && g(\mathbf{p}) \leq 0 \\ & && \mathbf{p} \in \mathbb{R}^{n_P} \end{aligned} \tag{4.49}$$

where $f, g : \mathbb{R}^n \rightarrow \mathbb{R}$ are locally Lipschitz functions.

The resulting optimization problem is in general non-smooth and non-convex when no further restrictions are imposed on $f(\mathbf{p})$ and $g(\mathbf{p})$.

A benefit of the formulation given in Equ. (4.49) is that the non-smooth optimization approach does not use Lyapunov variables. Lyapunov variables are dominated by the number of decision parameters \mathbf{p} which in the case of μ -synthesis are typically quite large. For further details on non-smooth optimization the references [7, 77, 10, 12] are recommended as a starting point.

Remark

The non-smooth optimization methods are implemented in the MATLAB Robust Control ToolBox since 2010. The routines are called ‘hinfstruct’, ‘looptune’ and ‘systune’. These are able to handle combined tunable blocks/controllers $K_i(\mathbf{p})$ to aggregate design requirements.

For high-dimensional problems, the use of initial values for the parameters \mathbf{p} is beneficial for accelerating the calculation. One possibility is the application of the classical method to synthesize \mathcal{H}_∞ controller, disregarding uncertainty. This state-space controller is reduced by truncation to the appropriate dimension and the parameter \mathbf{p} are initialized with the calculated values. But such a procedure may also lead to a local solution near the initialization as it is very likely that multiple local minima and maxima are prevalent. In general, it is rather recommendable to use random values for initialization with multiple runs and select the best result.

4.2 Modeling

For applying the \mathcal{H}_∞ -synthesis to the problem statement in the context of the AO setup, the individual components of the setup have to be thoroughly analyzed such that an appropriate system model for each component is identified. In this section, suitable model descriptions are identified and presented for each individual component of the given adaptive optics setup. In Fig. 4.7, the complete plant is shown in form of a block diagram, including the weighting matrices $W_d(s)$, $W_o(s)$ and $W_u(s)$, see Sec. 4.2.6 for tuning and shaping of the synthesized controller $K(s)$.

- $W_d(s)$ is the weighting function when the disturbance is frequency limited,
- $W_o(s)$ the weighting function for shaping the output and
- $W_u(s)$ weights the acting value for incorporating the limited frequency band of the actuators.

The dashed gray box (Fig. 4.7) is highlighting the physical components of the adaptive optics setup. $G(s)$ denotes the model for the DM (Sec. 4.2.2) as well as the

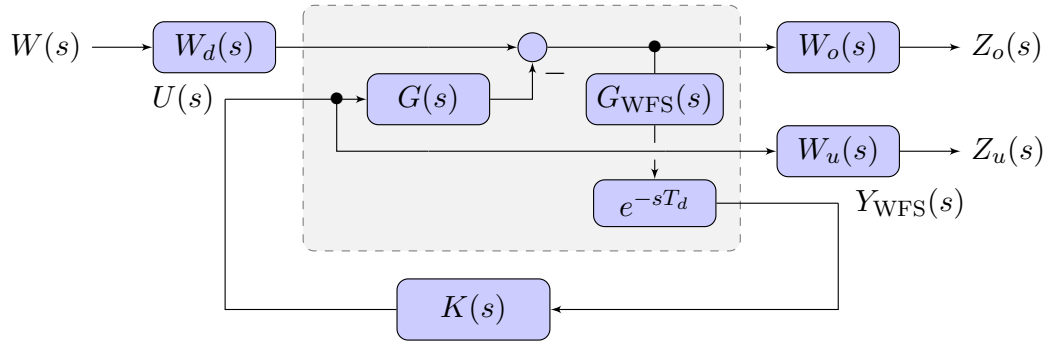


Figure 4.7: Block diagram of the overall system containing the weighting matrices $W_d(s)$, $W_o(s)$ and $W_u(s)$ and controller $K(s)$

tip-tilt mirror (Sec. 4.2.1). Depending on the purpose of the application, both or only one DM is used for the experimental setup. Additionally, tip-tilt and DM are treated separately as they are distinct components in the AO setup. The wavefront sensor is denoted with $G_{\text{WFS}}(s)$ followed by the delay e^{-sT_d} caused by the evaluation of the WFS (Sec. 4.2.3).

The real-time system, which is executing the controller and steering the complete setup, is not separately drawn as an individual block in Fig. 4.7 as this system can be supposed as perfect regarding its time behavior for most cases. The real-time system has been designed to match the requirements of the AO setup exactly, due to previous disappointing experiences with non adapted real-time systems. Nevertheless, the real-time system imposes some limitations on the complexity of the controller $K(s)$ and software development (Sec. 4.2.5).

In the following subsections, the physical characteristics of each component are considered as far it is helpful to develop a suitable model. Furthermore, several

assumptions are made to simplify the model identification and controller synthesis. These assumptions are thoroughly discussed and subsequently justified, as far as it is possible for the given case.

4.2.1 Tip-Tilt mirror

The employed tip-tilt mirror for the AO setup, is the S-330.2SL from Physik Instruments (PI). This tip-tilt mirror has a mirror surface diameter of $d = 31.5$ mm mounted on a cylinder (see Fig. 4.8). The tip-tilt mirror has already integrated strain gauges which are evaluated by the external proportional integral (PI) controller (PI E-509.X3). The actuating elements are piezoelectric stack actuators which are decoupled by design such that a change in tilt does not influence the tip deflection. The resolution of the mirror is specified to be 20 nrad and the maximum deflection is specified to 20 rad. The strain gauges are feed-back in the PI controller; afterwards, the desired signal of the PI controller is amplified by the amplifier (PI E-505.00) to drive the piezoelectric actuators. As the strain gauges are directly integrated in the calculation of the required voltage for powering the actuators, the linearity is enhanced by compensating the dynamic as well as possible hysteresis effects of the piezoelectric stack actuators.

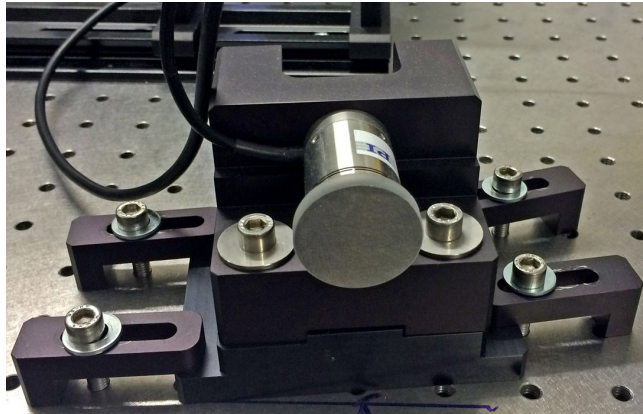


Figure 4.8: Image of the tip-tilt mirror S-330.2SL from Physik Instruments (PI)

For modeling the tip-tilt mirror, especially the transient behavior of the piezoelectric actuators, often a first-order plant is employed [80]. In general, this is applicable when neglecting the amplifier characteristics and no resonance frequency occurs or the utilized frequency band of the actuator is considerably below the first resonance frequency.

However in our case, the amplifier has also to be taken into account as we do not want to limit the performance in advance. Incorporating the amplifier can be automatically achieved when identifying the resulting closed-loop, combination of the PI controller and amplifier, as a black box. The system, then is used in a cascaded control scheme. That is, as long as the outer loop changes the setup-point much slower than the internal dynamics of the inner-loop, the identified model is valid.

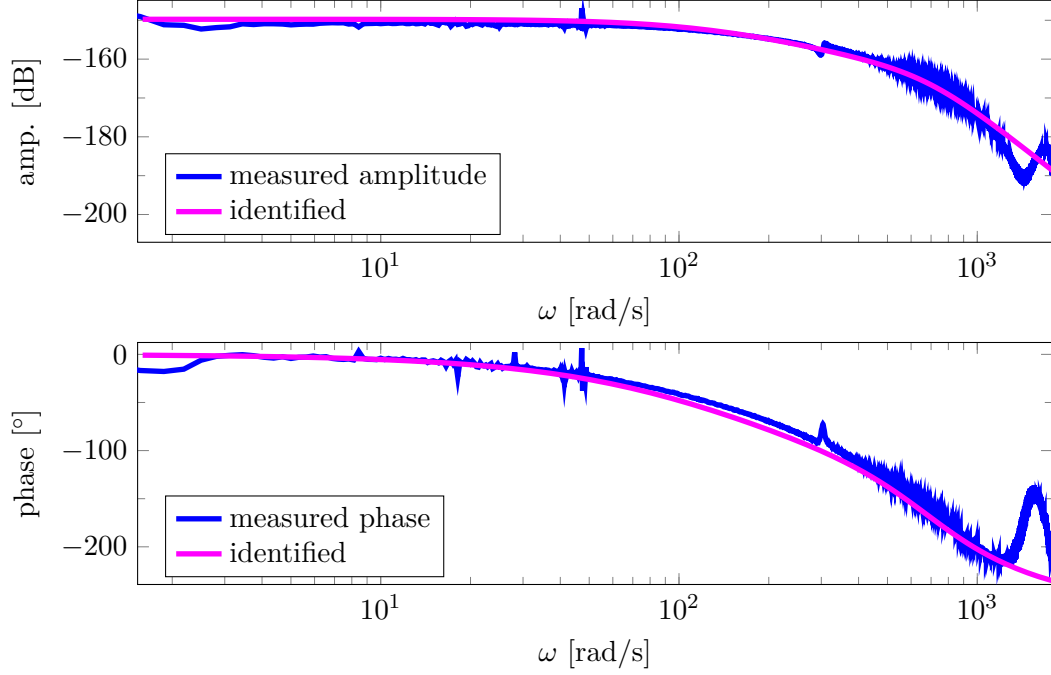


Figure 4.9: Bode plot of the measured data, superposed with the identified data for the tip-tilt mirror [68]

The frequency response (bode plot) of the tip-tilt mirror has been determined directly by using a scanning vibrometer PSV-400 from Polytec (see Appendix A.8 for more detailed information about the utilization of the scanning vibrometer).

In Fig. 4.9, the measured bode diagram of the tip-tilt mirror is given. An appropriate model of third order has been identified by application of the autoregressive with exogenous inputs (ARX) algorithm which is implemented in MATLAB system identification toolbox. Of course, other identification models could have been applied as well. Even by carefully analyzing the bode diagram in the desired range, the number and location of poles and zeros may be determined easily.

In our case, the transfer function has been determined as

$$G_{\text{tip/tilt}}(s) = \frac{514.9}{s^3 + 6432s^2 + 2.378e07s + 1.584e10}, \quad (4.50)$$

based on the measured frequency data incorporating the phase and amplitude.

The identified third order model, being identical for both degrees of freedom (tip- and tilt), is stable and minimum phase. The resonance frequencies, which are visible in the bode plot, are due to mechanical mounting reasons in the measurement setup. These resonances do not occur in the AO setup as the mounting has been constructed to suppress the undesired excitement. Nevertheless, the last resonant frequency in Fig. 4.9 at approximately $f \approx 1 \text{ kHz}$ or $\omega \approx 2000\pi \frac{\text{rad}}{\text{s}}$ is a resonant frequency which also occurs with appropriate attachment. This resonant frequency behavior has not been taken into consideration during identification of the model because the sensor rate (frame-rate) of the SHWFS is below the given frequency. Incorporating this

would just increase the dimension of the model for the tip-tilt mirror. Even if the frame-rate would be faster for another SHWFS, the assumption that the dynamics of the inner control loop is considerably faster than the outer loop would likely to be violated. The inner control loop is established with the devices from Physik Instruments (PI), so far, whose characteristics cannot be modified in this setup. Without the assumption problems are likely since the outer loop may then influence the inner loop and cause stability issues.

The transfer function given in Equ. (4.50) has not been normalized to one during the identification process, whereas in forthcoming steps, the normalized model will be used. The normalized model is adjuvant as the AIF will compensate for the amplification of the individual actuators (more details in Sec. 4.2.3).

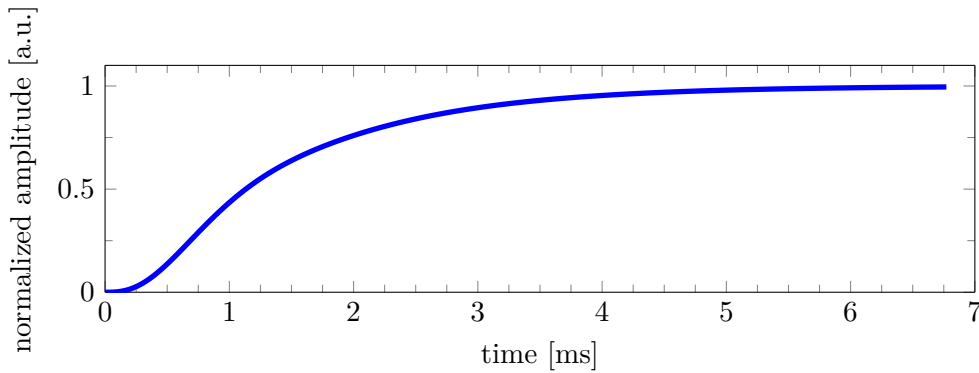


Figure 4.10: Normalized step response of the identified model of the tip-tilt mirror

To gain more insight into the time domain behavior of the tip-tilt mirror, Fig. 4.10 visualizes the transient behavior of the identified model when applying a step input. Additionally, several attempts to measure the deflection directly with the deflection decoder of the scanning vibrometer have been performed to supplement the frequency response measurements. But the deflection decoder turns out not suitable for step response measurements because the measurement drifts away with time. This drift likely occurs due to the integration of the noisy velocity signals to obtain the deflection in the deflection decoder. If larger deflections are measured, this is not crucial. However, for small deflections (hundreds of nm or a few μm), using the scanning vibrometer for the direct and accurate measurement of the deflection over time is not feasible.

Fig. 4.11 shows the measured step response by using the SHWFS. The red curve denotes the step signal whereas the blue and green curve are the reconstructed tip-tilt values. The delay in the response is due to the SHWFS and is discussed in Chap. 3. As the sampling time is slow, the ascent is not as in Fig. 4.10. In approximately 2-4 ms the stationary values is reached which closely coincide with the simulated step response.

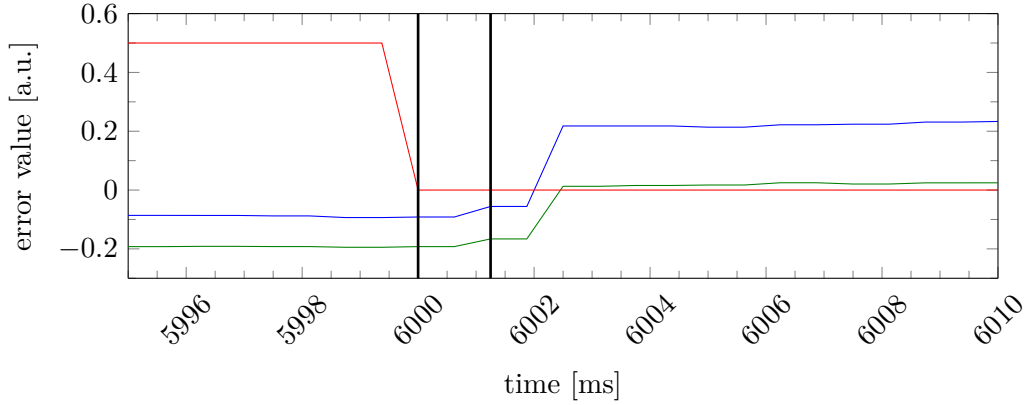


Figure 4.11: Step response of the tip-tilt mirror captured by the SHWFS; red curve denotes the step signal, green and blue curve are the reconstructed tip-tilt values, vertical black line marks the one step time-delay

4.2.2 Deformable mirror

The identification of the model for the DM is similar to the procedure for the tip-tilt mirror (Sec. 4.2.1). The basic principle and facts concerning the construction of the utilized DM are handled in Sec. 2.1.

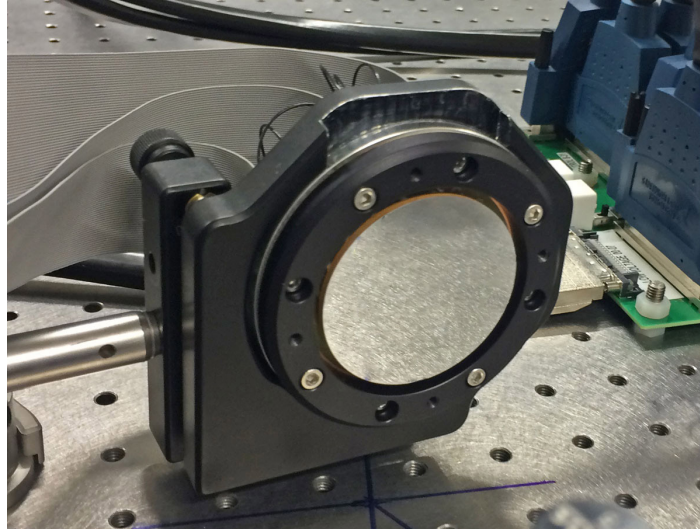


Figure 4.12: Deformable mirror from Fraunhofer Institute for Applied Optics and Precision Engineering (IOF) with 40 actuators having a diameter of 50 mm

However, instead of the presented mirror in Sec. 2.1, another mirror manufactured by Fraunhofer IOF has been utilized in lieu of the high-power mirror. The latter was not ready when starting the experiments since several technical difficulties during manufacturing and testing of the DM have had to be faced. Instead, a unimorph mirror with a 2 mm-thick polished glass substrate (borosilicate glass - B33) has been utilized [65]. As material for the piezo-actuators, an adhesively bonded piezoelectric disk (PIC 151, PICeramic GmbH) with $400\ \mu\text{m}$ thickness is used. The top electrode of the piezoelectric disk features in total 16 actuators outside the aperture. Inside

the aperture 24 actuators are given. The pie-slice actuator arrangement is visualized in Fig. 4.13b. The mirror is mounted and electrically contacted via 20 compliant cylinders on the rear surface of the mirror. The radius of the DM is 50 mm. All piezoelectric actuators share a common ground. This can be used to drive the actuators with ± 150 V while having a multichannel amplifier with an output of 0-300 V and another power supply with 150 V. The additional power supply is connected to the common ground for offering \pm voltage. Each actuator may be activated with 2 kV/mm, therefore in total 800 V can be applied. The deformable mirror with its mounting is depicted in Fig. 4.12.

The time domain identification (time-behavior) of the DM is, due to the same reason as for the tip-tilt case, not manageable with the scanning vibrometer PSV-400. Additionally, the response of the complete DM would be necessary for each time instance. As the scanning vibrometer measures each point individually sequentially, the measurements would be highly unreliable. Furthermore, the superposition of individual measurements in the time-domain is very sophisticated and prone to errors. In the case of the tip-tilt mirror, it would be only necessary to consider two degrees of freedom whereas in the case of the DM up to 40 different actuators are to be utilized which results in many degrees of freedom, in the best case exactly 40.

The frequency response of each actuator has been captured (see Appendix A.8 for more detailed information about the procedure). The grid for the measurement of the mirror surface has been adjusted to the actuator layout which is given in Fig. 4.13b. One important requirement is that the surface of the DM is captured well such that the individual deflection peaks are measured reasonably well. Otherwise, dominating measurement errors would lead to a mismatch in the modeling.

The identification step is divided into two different parts:

1. the steady state identification based on separate measurements
2. dynamic identification based on the previously acquired steady state value

The separation of the steady state and the dynamic identification is necessary because the resonant peaks of the deformable mirror are very large (see Fig. 4.14 for an example). Near the steady state the amplitude is very low compared with the neighborhood of the resonant frequency. Therefore, it is necessary to capture the steady state value very accurate. At the resonant frequency, the amplitude is almost one dimension larger which requires a higher range instead of higher resolution to prohibit clipping of the measurements.

Fig. 4.13a depicts the normalized static AIF of the DM. This representation is ideally suited to characterize the coupling of the actuators as well as a rough validation of the measurements. The procedure for the normalization is as follows: First, the primary switched actuator deflection is normalized to one. Then the other deflections are divided by that value. Therefore, the diagonal axis is thoroughly one in Fig. 4.13a. The white box in Fig. 4.13a visualizes the inner ring which corresponds with the first 24 actuators. It is clearly visible and easy to understand that the first eight actuators

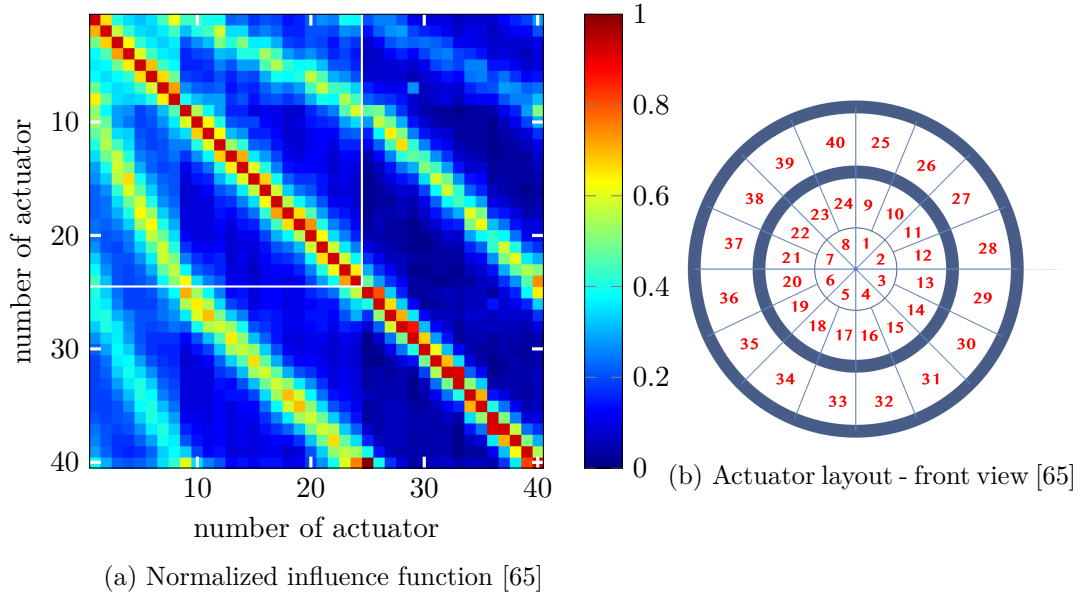


Figure 4.13: Visualization of the normalized static AIF and respective actuator layout/grid of the DM

are showing the strongest coupling. These eight actuators are directly connected with the minimum distance to each other. These couplings have 0.4 as the smallest value which corresponds with 40 % since this representation is normalized. Additionally, two/four side minor diagonals are prevalent which are the neighbored actuators of the corresponding actuator (see Fig. 4.13b). The minor diagonals are not totally parallel to the diagonal axis as the individual inner circles have different quantities of actuators.

Nevertheless, the overall minimum coupling is greater than 20 %. Thus, even in the static case it is likely that a lot of control input is required to compensate for the coupling to reject disturbances. However, it is worth mentioning that a local coupling of the actuators is highly desirable for obtaining a smooth surface when measuring with the SHWFS.

At the beginning of the identification process and adjustment of the AO setup no other DM has been available. Furthermore, it was a requirement to use an in-house mirror from Fraunhofer IOF. For some time, different DMs have been designed and produced by Fraunhofer IOF which produce a larger stroke. As a side-effect due to e.g. changes regarding the mirror substrate, the dynamic behavior of these latest DMs have also changed significantly. Thus using a newer DMs would require to perform the identification process again. The applied mirror for the identification and experiment in this work has a rather thick mirror substrate resulting in the observed large coupling and a high first resonant frequency.

The employed DM has a rather low stroke/deflection of $\approx 500 - 600$ nm. Due to the low deflection of the DM, even ≈ 50 nm measurement inaccuracy or noise, which possible arise when using an SHWFS for measuring the wavefront, correspond to a relative error of ≈ 10 % of the maximum deflection of the employed DM.

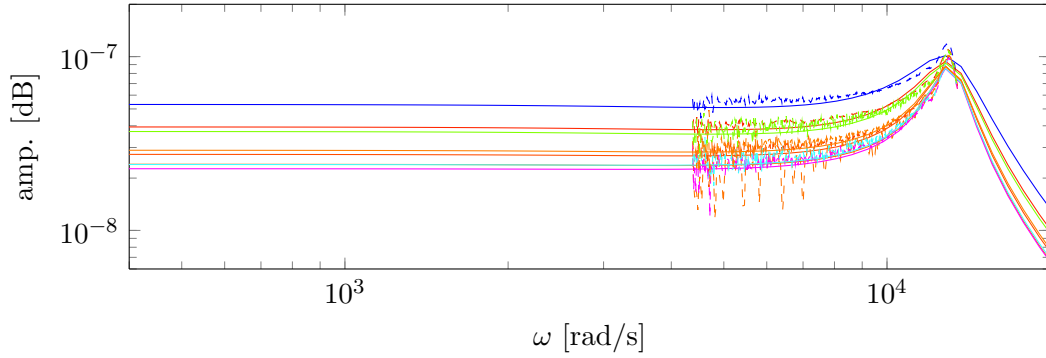


Figure 4.14: Measured and identified values for actuator one to eight when deflecting actuator one with an amplitude of 30 V

In Fig. 4.14 some exemplarily measurements of the frequency response are given. The highest amplitude (colored with blue) corresponds to actuator one. Plotted in solid line, the calculated curves are shown, whereas the dashed curves are the actually measured ones. The most important fact is that up to about $3100 \dots 3800 \frac{\text{rad}}{\text{s}}$, the behavior is quasi-static. Additionally the resonant peak cannot be completely captured by the given third-order model given in

$$F(\omega) = F(0) \frac{\frac{1}{p_1}j\omega + 1}{\left(\frac{1}{p_2}j\omega + 1\right) \left(\left(\frac{j\omega}{p_3}\right)^2 + 2\frac{p_4}{p_3}j\omega + 1\right)}. \quad (4.51)$$

Nevertheless, due to the relative low sampling rate of the SHWFS with ≤ 900 Hz, a precise description in the higher frequency range is unnecessary and would only increase the model complexity.

In spite of a sufficient representation in the high frequency range, it is essential that the resonant behavior is approximated well since the change in amplitude is quite large. Apart from that it is important to suppress undesired high frequency actuation. More resonant peaks might be measured at higher frequencies, in general, integer multiples of the first resonant peak will also be resonant frequencies.

The third order model is derived by the typical resonant behavior of the piezoelectric actuators as well as the characteristic of the driving amplifier which tends to act as a low-pass filter. Thus, three poles and one zero are identified for each measurement. Of course, higher-order models may be derived, however, the third order model is sufficient for the required accuracy and frequency range.

A non-linear least-square optimization algorithm has been used to identify the parameters p_1, \dots, p_4 according to Equ. (4.51). If p_1 and p_2 are close to some boundary then this zero/pole is removed for its low influence. $F(0)$ is the steady state value which is determined beforehand, based on extra measurements with higher accuracy. Using the frequency range of $1200 \dots 3100 \frac{\text{rad}}{\text{s}}$, the steady state value is determined by the median. Frequencies below $1200 \frac{\text{rad}}{\text{s}}$ corresponding to $\frac{1200}{2\pi} \text{ Hz} \approx 190 \text{ Hz}$ are superposed by e.g. the vibrations of the building and cannot be well captured with the

scanning vibrometer. Separate measurements with even higher accuracy have been conducted in this range to be sure that there is no unexpected behavior. As this was not the case, these values have been dropped in the identification process.

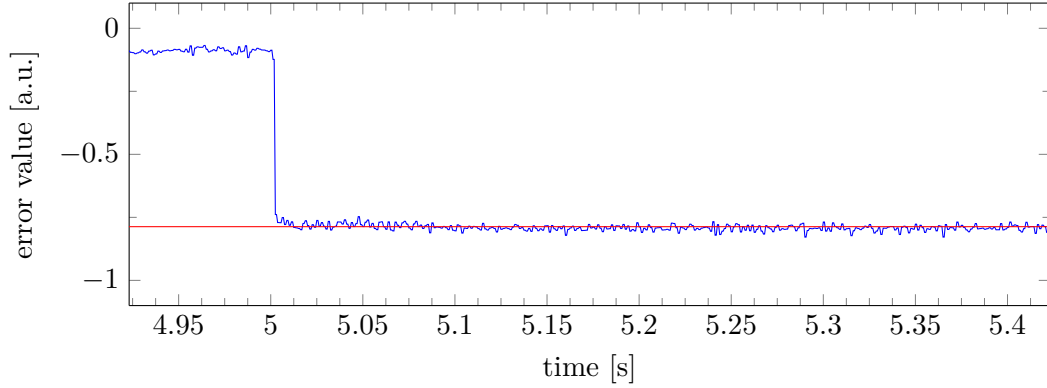


Figure 4.15: Measured step response of one piezoelectric actuator via sampling with an SHWFS, red line denotes the stationary value and blue curve the actual measured reconstructed actuator value through the control matrix

To check the identified model some time-domain measurements have been carried out with the SHWFS. Fig. 4.15 is showing the reconstructed recorded actuator value when a step is applied to one actuator. Based on this figure it is obvious that between zero to 400 ms after the step has been applied, no creep behavior is present. Fig. 4.16 is zoomed in time to be able to better assess the behavior in the near distance of the step whereas the red curve denotes the applied step. The delay between the blue and the red curve is due to the processing of the SHWFS data in the FPGA. Thus, a delay of one frame (in this case $\frac{1}{850}$ s) of the SHWFS is present (see Chap. 3).

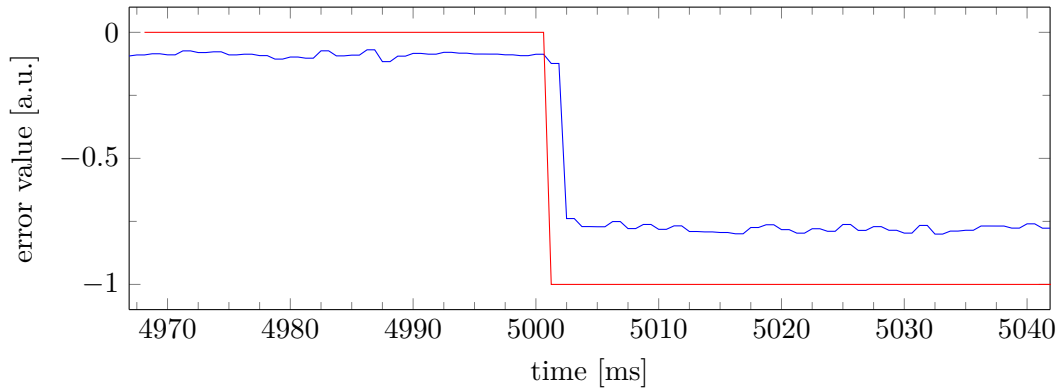


Figure 4.16: Fig. 4.15 partially zoomed such that the behavior around the time of the step is better visible

When comparing Fig. 4.17 with Fig. 4.16 it is apparent that the simulated step response does not really match with the recorded at first sight. As the sampling rate of the SHWFS is below 900 Hz, the overshooting is not visible in the recorded data of the SHWFS. Additionally, the existing time-delay has to be neglected for comparison. When looking at the required time for reaching the stationary value, the time of the

recorded and simulated step-response almost is identical. Therefore as far as it is possible by using the recorded data from the SHWFS, the identified model matches with the measurements well.

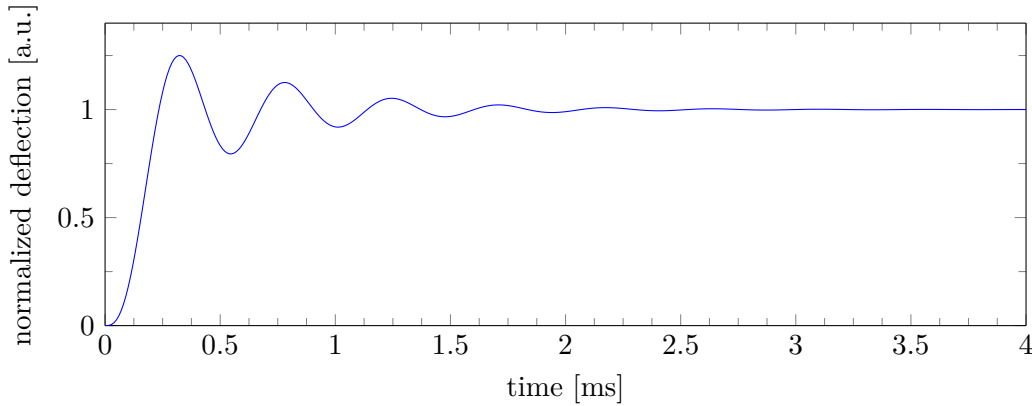


Figure 4.17: Normalized step response of the identified model for the DM

Lastly, piezoelectric actuators in general show some hysteresis. Fig. 4.18 visualized the hysteresis measured by the SHWFS. The hysteresis is not excessive but it is obvious that it is not symmetric regarding the origin. This results from gathering the control matrix by using the AIF and some long term creep behavior in the range of seconds. The AIF is captured with positive voltages (thus a positive deflection). Piezoelectric actuators do not behave the same when having positive or negative voltage regarding deflection. Furthermore, the rather low stroke of the DM and the limited resolution of the SHWFS complicate to obtain precise measurements. But the hysteresis is considered during synthesis by approximating it as a time-invariant uncertainty.

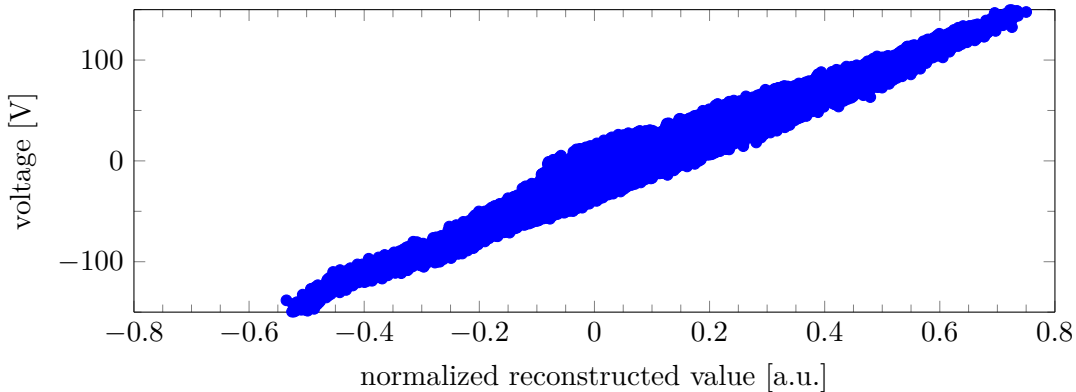


Figure 4.18: Measured hysteresis based on SHWFS measurement and reconstruction through control matrix

The identified model of the DM is not presented in numeric terms here as the model has 40×40 transfer functions and would require a lot of space and the benefit of numeric values would be questionable. The identified model can be found on the enclosed CD-ROM (see Appendix A.10).

4.2.3 Wavefront sensor

Some different principles of wavefront sensing have been presented in Sec. 2.2 and the associated evaluation algorithms have been depicted and discussed in Chap. 3.

When involving the SHWFS as a sensor in the control loop, either the information of the slopes or other representations such as Zernike coefficients are used. In principle, the reconstructed wavefront itself may be used for this task as well. Each option has its own advantages. For example, using the slopes requires less computational power, whereas using the reconstructed wavefront may suppress noise which occurs due to the measurement method itself. Zernike coefficients may be used to control only specific modes of the DM, i.e. controlling only defocus or disregarding tip-tilt behavior. So, the representation of the wavefront highly depends on the task.

For a high-performance adaptive optics setup the direct use of the slopes approach is beneficial because other methods, such as reconstruction of the wavefront, require additional time for their inherently more complex calculations. The slope values have to be correlated with the required values for the actuator of the individual mirrors. When combining the slopes Θ_v in a column vector, the resulting vector multiplied with the control matrix Γ produces a vector of commands u . This vector represents the required actuating value of the DM to achieve the exact deformation. Applying the calculated command values to the mirror (additionally to the previous applied commands because the plant does not have an integral part), the result is a minimum RMS wavefront error [114].

The control matrix is the pseudo-inverse of the measured poke matrix denoted as AIF matrix. The poke matrix is a matrix which measures the influence of each actuator by using the slopes information of an SHWFS. Usually, each actuator is deflected by a fixed voltage, whereas the resulting slopes are stored into the corresponding column of the poke matrix P [114, 63]. Thus, the resulting matrix (poke matrix) describes the relationship between commands applied to the DM and the slopes of the SHWFS.

This relation may be expressed as

$$\Theta_v = Pu \quad (4.52)$$

where Θ_v is the vector of the wavefront slopes, P the poke matrix and u is the vector of commands applied to the mirror. The command to the mirror does not have to be necessarily a voltage since DMs do not always have a linear relationship between the deflection and the voltage. Then, a LUT can be used for example to modify the mapping of desired deflection to the required voltage. The control matrix Γ may be calculated based on the poke matrix P by calculating the pseudo-inverse of P .

Therefore, the command voltages result from

$$u = \Gamma \Theta_v \quad \text{with} \quad \Gamma = (P^T P)^{-1} P^T. \quad (4.53)$$

For simplicity and due to computational reasons [63], Γ is computed in advance such that only a matrix multiplication of Γ and Θ_v must be performed for each measurement of the SHWFS online. This is manageable as the poke matrix P is determined before the control-loop is activated and thus does not change afterwards. The dimension of the matrix multiplication is given by $\Theta_v \in \mathbb{R}^{n_S \times 1}$ and $\Gamma \in \mathbb{R}^{n_A \times n_S}$, whereas n_S is the number of slopes in x- and y-direction and n_A the number of actuators of the mirror. In the case of the Imagine Optics HASOTM3 Fast SHWFS $n_S = 196 \cdot 2 = 392$. The required time for the calculation can be roughly approximated by counting the number of required multiplications which are $392 \cdot 40 = 15680$. The number of multiplication are comparable with a square matrix multiplication of $n = 25$. Therefore, when analyzing the time consumption of a $n = 50$ square matrix multiplication, the time consumption may be neglected as the time is below $10 \mu\text{s}$ (see Appendix A.6.1).

In the presence of additive noise for each SHWFS measurement the formula given in Equ. (4.53) may have a large error covariance matrix [45, 73, 35]. The measurement noise of the SHWFS can be converted into an equivalent stroke. Thus, it is obvious that a large error covariance matrix will likely occur when the stroke of an actuator is not large when compared to the equivalent noise stroke. With techniques such as averaging the influence of the measurement noise may be decreased but this requires more time during calibration. This circumstance has been analyzed in detail [45]. As a solution, the Hadamard matrix is suggested as actuating pattern. This special actuating pattern has a large determinant compared to the diagonal matrix which is indirectly used for the first approach for gathering the poke matrix.

For the given AO setup, the required calibration time of the system is not as important as when e.g. calibration should be performed online for space applications with hundreds or even thousands of actuators. Thus, the application of the Hadamard calibration can be beneficial but averaging is also practicable solution.

An often used approach is to perform some kind of coordinate transformation for the slopes, e.g. using the Zernike polynomial set (see Sec. 3.1). The Zernike polynomials are used to describe the wavefront whereas the slopes are the derivative of the wavefront. The reason of such coordinate transformation is that measurement noise always occurs. This noise can, at least partially, be suppressed by means of fitting the coefficient of the given polynomials. However, for discrete data the Zernike polynomials are no longer orthogonal. Moreover, the derivatives of Zernike polynomials have no orthogonality property, regardless of having discrete or continuous data. However, a benefit when using the Zernike set is that special modes may be inhibited, e.g. the tip-tilt mode of an DM. The number of coefficients fitted can be adjusted by reducing the number of coefficients to describe the wavefront. A shift, resulting in a tip-tilt error (described in Sec. 3.3.6) would not be problematic if the derivatives of the given polynomial set would be orthogonal. Notwithstanding, when using unmodified Zernike polynomials this property is not fulfilled.

The complexity of transforming the slopes to Zernike polynomials is not high (see

Sec. 3.1). Performing the calculation online, only requires $28 \cdot 392 = 10976$ multiplications, when $n_Z = 28$ denotes the number of Zernike coefficients and $n_S = 196 \cdot 2 = 392$ the number of slope values. Comparable with the direct conversion of slopes to actuator values, the time consumption is negligible (see Appendix A.6.1).

Afterwards, however, the Zernike coefficients have to be converted into actuating values which requires then requires $28 \cdot 40 = 1120$ multiplications with $n_Z = 28$ and $n_A = 40$.

Fig. 4.19 gives a short overview regarding the required evaluation time of the SHWFS. The main difference between the time diagram in [65] and Fig. 4.19 is that the diagram shown here also includes the matrix repositioning step from Sec. 3.3.6. The adaptive thresholding does not increase the time consumption. As a result, this step is not visualized in the corresponding Fig. 4.19.

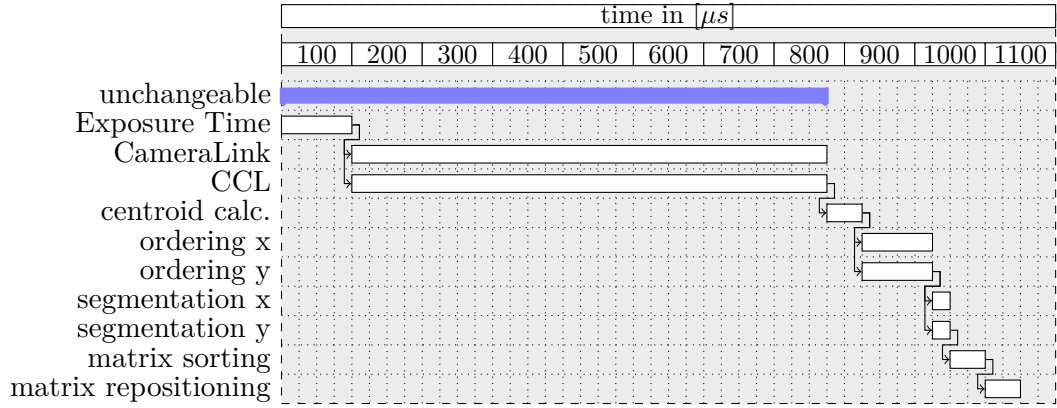


Figure 4.19: Time evolution (each step rounded up to $25 \mu s$) during SHWFS evaluation by application of the FPGA approach

After all a time delay of $\tau = 1050 \mu s$ prevails for the SHWFS model. In total, the model for the SHWFS can be denoted as

$$G_{WFS}(s) = e^{-1050 \cdot 10^{-6} s} \Gamma \Theta_v \quad (4.54)$$

where the required time for performing the calculation of Equ. (4.53) is not included. This calculation given in Equ. (4.53) is performed on the real-time computer (see Sec. 4.2.5) and may be neglected as the complexity is rather low (see Appendix A.6.1).

When also taking the sampling rate into account the delay will automatically be a multiple of the sampling rate of the controller. The trigger input for starting the exposure of the camera image of the SHWFS is set with the sampling rate of the controller. This implies that when having a main sampling rate of 1700 Hz and every second the trigger for the SHWFS is set then the delay is $\frac{1000}{850} \text{ ms} = 1176.5 \text{ ms}$. If the trigger rate would be only 800 Hz then the delay would increase to 1250 ms as long as the sampling rate is 1600 Hz.

Nevertheless, increasing the sampling rate also requires that the controller signals are calculated at that specific sample rate. In others words, the sampling rate is limited

by the complexity of the controller and the processing power of the corresponding real-time system.

Mixed optimization approach for control signal reconstruction

The standard approach of control signal reconstruction is using the pseudo-inverse of the poke matrix P to minimize the \mathcal{L}_2 -norm of the RMS error. In general, this procedure is ideal to achieve best performance regarding the total error (in terms of RMS error) but can lead to large values for the control signal.

When the control signal is subject to output limits, the approach to minimize the error while having limits for the control signal in parallel can be very handy. One straightforward idea is to incorporate the \mathcal{L}_1 -norm of the control signal when minimizing the \mathcal{L}_2 -norm of the wavefront error. This approach is based on a mixed goal optimization for reconstruction of the control signal. One problem while performing this approach is that no longer the pseudo-inverse of the poke matrix P can be applied. Instead, the optimization has to be done online with the impediment that the problem is no longer convex. Therefore, it requires much more computational effort, likely being non-deterministic, can lead to immense delays and thus, to a reduction of the overall sampling time and an increase in the delay. The feasibility of the approach has been analyzed the results of which are given in Appendix A.6.7.

The following optimization problem needs to be solved for each time instance

$$\min_{u[k+1] \in \mathbb{R}^{n_A}} \left(\sqrt{(\Theta_v[k] - Pu[k+1])^T (\Theta_v[k] - Pu[k+1])} + \beta \sum_{n=1}^p |u_n[k+1]| \right) \quad (4.55)$$

where $\beta = 6$ has been chosen for the specific feasibility study. By adapting β , a trade-off between minimizing the control signal and RMS error may be made.

If for example β is chosen to be equal to zero then the optimization will have the same result as Equ. (4.53). But Equ. (4.53) only requires a matrix multiplication instead of solving an optimization problem online, thus, will be much faster.

Due to the equivalence of p-norms, $\|\cdot\|_1$ can also be written as

$$\|x\|_1 \leq \sqrt{n} \|x\|_2$$

whereas n is the dimension of x . Thus the optimization problem can be simplified but the result will not be the same due to the inequality. Therefore the benefit is not given as n is rather huge and only an upper bound will be received.

The result of the feasibility study is that even when not considering the control signal limits, the required time for performing such an online optimization exceeds the intended time line (see Appendix A.6.7). Thus, the approach limits the overall sampling time and using an online optimization would lead to worse performance in terms of the overall speed.

4.2.4 Delay handling

In many textbooks and the corresponding theory, time-delay system are rarely treated. Nevertheless, nearly every system shows some time-delays, e.g. due to evaluation of sensor information or communication delays. Thus, even when considering an LTI systems, time-delays are commonly present. However, depending on the dynamics of the system, time-delays may often be neglected because the time-delay has a small impact within in the whole context.

For the utilized AO system, the different time-delays of the system are elaborately examined for the real-time system (Appendix A.6). Sec. 3.4.3 and 3.4.2 have already discussed the time-delay for the evaluation of the SHWFS. In Sec. 4.2.5 the different time-delays are summarized such that a controller for the AO system can be synthesized. In the context of high performance AO, the identified time-delays cannot be ignored when acceptable performance results shall be guaranteed.

A constant time delay $\tau > 0$, i.e. a shift in time $y(t) = u(t - \tau)$, in the Laplace-domain results in

$$Y(s) = e^{-\tau s}U(s) = H(s)U(s).$$

The time-delay, represented by $H(s) = e^{-\tau s}$, is a regular transfer function. Nevertheless, even if the time-delay may be represented in frequency domain the difficulty is that the representation is not rational since it shows an infinite number of poles and zeros. Using regular \mathcal{H}_∞ or \mathcal{H}_2 methods for LTI systems, rational transfer functions are required.

Therefore, one may approximate the time-delay $H(s)$ or reformulate the problem such that standard methods may again be applied.

Padé-approximation

A classic approach is to use a series approximation such as the Taylor- or Padé-approximation. The latter has better characteristics for the delay case, e.g. being more accurate or of the same relative degree. The general Padé-approximation for the time-delay may be written as follows

$$e^{-\tau s} = \frac{1 - k_1 s + k_2 s^2 + \dots + (-1)^n k_n s^n}{1 + k_1 s + k_2 s^2 + \dots + k_n s^n}, \quad (4.56)$$

where n denotes the order of the approximation. Several characteristics can be derived based on Equ. (4.56), e.g. the numerator has different signs of its coefficients which means that at least one zero has to be in the right-half complex plane.

Choosing $n = 3$ as the order of the approximation, the following representation is obtained.

$$e^{-\tau s} \approx \frac{1 - \frac{\tau}{2}s + \frac{\tau^2}{10}s^2 - \frac{\tau^3}{120}s^3}{1 + \frac{\tau}{2}s + \frac{\tau^2}{10}s^2 + \frac{\tau^3}{120}s^3}$$

Applying the Hurwitz' criterion, it can easily be shown that all poles are in the left-half complex plane, thus, the transfer function is stable.

The order of the Padé-approximation is selected according to the desired accuracy and the concrete time-delay τ itself. Requiring a good approximation at higher frequencies necessarily leads to higher order of the Padé-approximation.

We may consider that the time-delay τ is uncertain and varies from experiment to experiment, but staying constant for each individual experiment. For example, when the time-delay is nominal 3.5 ms but may also be 3-4 ms then the time-delay may be written as:

$$\begin{aligned}\|\tau_1\|_\infty &\leq 1 \\ \tau &= 0.0035 + \frac{\tau_1}{2} \cdot 10^{-3}\end{aligned}\tag{4.57}$$

The case of unknown time-delay without being time-varying may occur e.g. when two systems have the exact same frequency but their phase between each other is unknown. Let system one be the real-time controller and system two the sensor. System two has a nominal time-delay of 3 ms due to signal processing. Depending on the exact time of the start of each system, the resulting time-delay can be any value between 3-4 ms. In other words, a phase shift between the two systems is existent. Equ. (4.57) may be treated as a parametric uncertainty as dealt with in Sec. 4.1.2. The μ -synthesis approach may handle such a parametric uncertainty. The non-smooth μ -synthesis can synthesize a controller of given order which minimizes the \mathcal{H}_∞ norm of the closed-loop system under consideration of the unknown time-delay [66].

Delay as norm-bounded uncertainty

An alternative to approximating the time-delay is to reformulate the time-delay as norm-bounded uncertainty without using the Padé- or Taylor-approximation. This method has been presented in [57]. The main difference, compared with the previous approach, is that the time delay is now restricted to 0-4 ms. In the former approach, the time-delay was considered in the range between 3-4 ms.

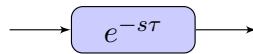


Figure 4.20: Time-delay block in the frequency/Laplace domain

Starting from an ordinary time-delay in the Laplace domain (Fig. 4.20) the delay block may be redrawn as shown in Fig. 4.21. So far, no approximation or restriction appears and both representations are totally equivalent.

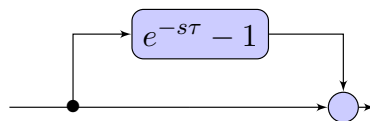


Figure 4.21: Time-delay block from Fig. 4.20 redrawn

Analyzing $e^{-s\tau} - 1$, one straight forward approach is to split $e^{-s\tau} - 1$ in a norm-bounded uncertainty with $\|\Delta\|_\infty < 1$ and a transfer matrix $W_u(s)$ which is depicted in Fig. 4.22. Due to this transformation and separation the delay model has now become rational and approximated at the same time. However, $W_u(s)$ has to be chosen appropriately for a good approximation. $W_u(s)$ is an upper bound for a given time-delay τ_u . Therefore every time-delay between $0 \leq \tau \leq \tau_u$ is implied in Fig. 4.22.

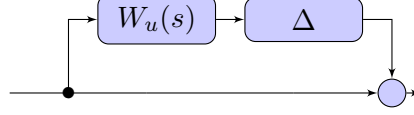


Figure 4.22: Time-delay redrawn using a norm-bounded uncertainty

The chosen weighting matrix is as follows

$$W_u(s) = \frac{2s \left(s^2 \frac{\tau^2}{4} + \left(\tau + \frac{\tau}{4} \right) s + 1 \right)}{\left(s + \frac{2}{\tau} \right) \left(s^2 \frac{\tau^2}{4} + \tau s + 1 \right)}. \quad (4.58)$$

Fig. 4.23 is showing $W_u(j\omega)|_{\tau=4\text{ ms}}$ as well as $e^{-0.004j\omega} - 1$ and $e^{-0.003j\omega} - 1$ for the given frequency range. It is obvious that $W_u(j\omega)|_{\tau=4\text{ ms}}$ is the upper bound which incorporates all time-delays from 0-4 ms.

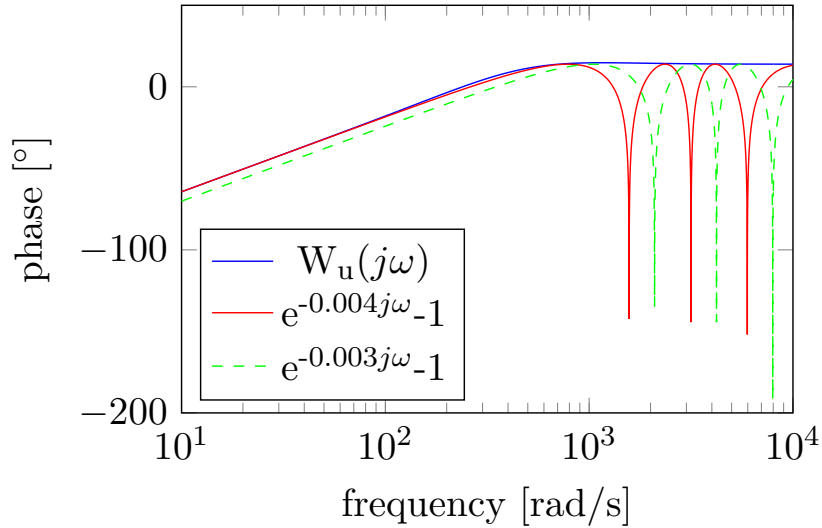


Figure 4.23: Phase diagram for $\tau = \{3\text{ ms}, 4\text{ ms}\}$ and transfer function $W_u(j\omega)|_{\tau=4\text{ ms}}$

The AO setup has been constructed such that no unknown time-delay can appear, as it was possible in the first AO setup employed [66]. Therefore, the time-delay may be approximated with the standard Padé-approximation.

4.2.5 Real-time system

In general, the real-time system is very crucial for achieving high performance as the system limits the manageable complexity. Also, the system can introduce delays

and other side-effects which will likely degrade the achievable performance. Thus, the real-time system has to specially selected and some particular parts need to be supplementary developed for the experimental setup. There are almost no limitations regarding reasonable controller dimensions. Additionally, the system is tuned to introduce almost negligible time-delay.

For more detailed information concerning the real-time system, Appendix A.6 is recommended where several experiments are listed with the corresponding results, e.g. on experiments regarding latency and performance of the real-time system.

The real-time system is required to perform essential calculations, e.g. the conversion of the slopes to actuating values (see Sec. 4.2.3). Computing the controller signals and writing the corresponding analog values to the outputs such that the amplifier of the actuators can drive the actuators is part of the real-time system as well. Several other activities are performed by the real-time system, however, these are not essential for closed-loop operation; e.g. storing individual signals for offline and control behavior analysis.

In view of the experimental results no further dynamic model is required.

4.2.6 Weighting matrices

An appropriate choice of the weighting matrices is crucial for a successful application of \mathcal{H}_∞ methods such that acceptable performance of the closed-loop dynamics is accomplished. Different weighting matrices should be applied when using e.g. the mixed-sensitivity approach. The weighting of the output $W_o(s)$ (Fig. 4.7) or generally speaking the error itself, imposes performance specifications as well as closed-loop quality[79].

The weight $W_o(s)$ typically is a square diagonal matrix of the form

$$W_o(s) = \text{diag} (W_{o11}(s), \dots, W_{o22}(s), \dots, W_{oqq}(s)) \quad (4.59)$$

whereas q is the dimension of the outputs.

The inverse of the different elements W_{oii} shape the sensitivity function $|Z_{oii}(j\omega)|$ as an upper bound, where $Z_{oii}(s)$ denotes the i -th element of the diagonal of $Z_o(s)$.

Typically, W_{oii} is of the following form

$$W_{oii}(s) = K_i \frac{\alpha_i s + 1}{\beta_i s + 1}. \quad (4.60)$$

Due to the nature of the method for synthesizing the controller $K(s)$, the resulting controller decouples the outputs such that, in a good approximation, $Z_o(s)$ is a diagonal matrix. In general, the behavior of $W_o(s)$ is a low-pass filter of first or second order. Increasing the order of the weighting matrix is not beneficial because the inverse of the weighting matrix shape the resulting sensitivity matrix.

Higher order filters with -40 dB per decade or less will tend to deteriorate the performance as the gain margin will decrease as well as the phase margin. If the decay is

lower than -40 dB at the cross-over frequency then there is no phase margin anymore for an LTI systems.

From an analysis of a second-order systems it is obvious that a small phase margin will lead to overshooting in the step response. On the contrary, widely increasing the phase margin will result in creeping slowly to the steady state value. This statement is applicable for second-order systems, but also may be transferred to higher-order systems with some restrictions.

The general problem of a small gain margin is that uncertainties in parameters may lead to instability easily. Thus, these uncertainties should be modeled and considered in Δ with the consequence that they are not apparent anymore in the rest of the model.

Most of the characteristics for $W_o(s)$ apply to $W_u(s)$ as well. $W_u(s)$ is weighting the actuating values (see Fig. 4.7). The typical shape of $W_u(s)$ is a high-pass filter to prevent the appearance of high-frequency signals because of the limitation of the actuator bandwidth. Other reasons are the discretization step of the controller in the case that the controller is realized in discrete time. Thus, when having a low sampling time, high frequency oscillations must be inhibited as these cannot be seen by the controller due to the low sampling rate and aliasing effects can occur.

The weighting matrix $W_d(s)$ for weighting the input can be used if it is known in advance that the disturbances are specifically shaped in the frequency band. In most cases, due to the arising complexity of the problem, $W_d(s)$ is replaced by a diagonal identity matrix, thus, may even be totally removed.

5 Experimental Setup

Controlling a laser with Linux is crazy, but everyone in this room is crazy in his own way. So if you want to use Linux to control an industrial welding laser, I have no problem with your using PREEMPT_RT.

LINUS TORVALDS: (2007) at *Kernel development*

This chapter is devoted to the experimental setup and the achieved results. To this end, the individual components i.e. the optical breadboard setup, real-time system, real-time wavefront evaluation and synthesized controller are combined all together. The optical breadboard setup has already been presented in Sec. 2.3, whereupon the focus had been on the concept, the optical setup as well as on the individual optical elements for diameter beam adjustment. The proposed setup is an enhanced version of the original setup which has been outlined during KD OptiMi - Kompetenz Dreieck Optische Mikrosysteme (KD OptiMi)¹.

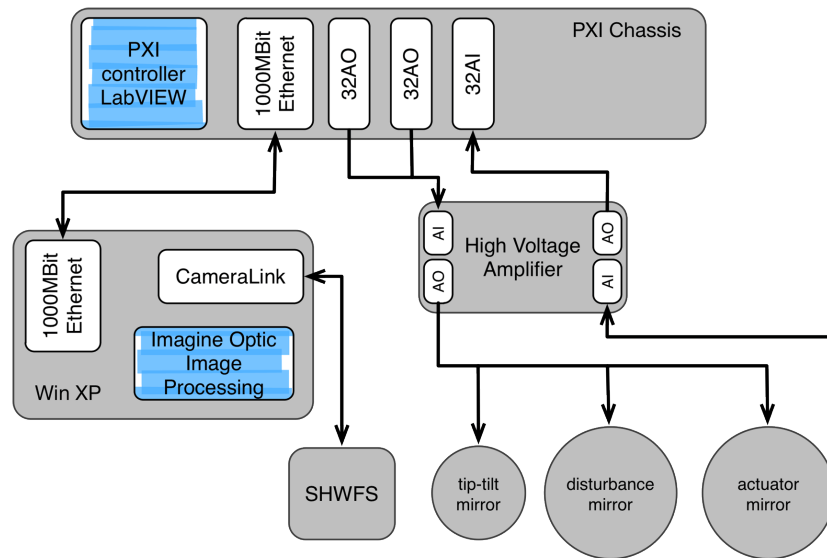


Figure 5.1: Overview of the LabVIEW based concept used for KD OptiMi

The system, visualized in Fig. 5.1, has been outlined in phase one of KD OptiMi, previous to the launch of the work presented in this dissertation. The basis has been a laboratory virtual instrumentation engineering workbench (LabVIEW) PCI extensions for instrumentation (PXI) system, running LabVIEW Real-Time Module

¹ KD OptiMi has been originally funded by the Bundesministerium für Bildung und Forschung (BMBF) in 2008 to 2010 for phase one and from 2011 to 2013 for phase two, see <http://www.optimi.uni-jena.de/>.

equipped with several build-in cards such as analog inputs, analog outputs cards and separate ethernet card.

The Shack-Hartmann wavefront sensor (SHWFS) is connected to a standard Windows computer running Windows XP over a slot-in CameraLink interface card. The personal computer (PC) evaluates the camera image which is transferred via the CameraLink card and evaluates the wavefront by using the provided software from Imagine Optics. The results are either spots, slopes or even the reconstructed wavefront. In this case, the spots are transmitted with a LabVIEW program which itself is communicating via Transmission Control Protocol/Internet Protocol (TCP/IP) (or optional user datagram protocol (UDP)) with the main PXI system. The PXI system has in total 64 analog outputs and 32 analog inputs which are used for controlling the actuators and reading e.g. the temperature sensors and strain gauges of the deformable mirror (DM), if available. In the first concept developed during KD OptiMi, the deformable mirror has been designed to include strain gauges [36, 37], temperature sensors and heating elements additional to the piezoelectric actuators. The analog signals are fed into the high voltage amplifier with 64 channels (HV64) for signal level adaption such as filtering and level adjustment.

Due to different technical reasons, DM manufactured by Fraunhofer Institute for Applied Optics and Precision Engineering (IOF) do not incorporate all functionality in a single device yet. For the work outlined in this dissertation, thus, a DM is used which only has piezoelectric actuators (see Sec. 4.2.2). Notwithstanding it should be noted that Fraunhofer IOF has built DMs which are capable of handling high power laser beams up to 6.2 kW [92, 90] successfully.

Several experiments and tests have been carried out with the initially proposed LabVIEW based system. The results have shown that the individual components and their interaction do not lead to good results. One problem which has been identified is the evaluation of the SHWFS on the Windows PC and the transmission of the data via ethernet. Due to the lacking real-time behavior of Windows, the resulting time-delay is unknown and in-fact also proved to be even time-varying [68, 66].

What is more, the performance of the complete system has not been acceptable, resulting in insufficient time for computing medium up to more complex controllers for closing the loop (closed-loop operation). Firstly, a lot of effort has been spent optimizing the whole setup in order to improve performance. For this reasons, preliminary results have been gathered with this setup [68, 66], but the delay of the black-box wavefront evaluation has remained time-varying due to the operating system (OS) and the network communication.

Supplemental to the hardware limitations there has also been software issues. The implementation of the control algorithms has been quite complicated with the LabVIEW based system because the controller was synthesized in MATrix LABoratory (MATLAB). Based on the controller, a Simulink model has been created to simulate the adaptive optics (AO) system in closed-loop mode. Using the created Simulink model, a dynamic link library (DLL) is generated by application of Simulink Coder.

This DLL is then integrated in the LabVIEW program for the control purpose and for managing the experimental setup. This approach is not very gentle regarding the use of available resources because the overhead from the inclusion of the DLL in LabVIEW is quite large. Additionally, the synthesized C-code is not optimized for the given LabVIEW system architecture, thus, is not exploiting the possibilities given by the system. It is clear that e.g. a state-space controller can be integrated directly into LabVIEW by using the Hybrid Graphical/MathScript Approach², but copying the respective numbers manually is error prone. The identification sequence for the actuator influence function (AIF) would be required to be coded twice (in MATLAB and LabVIEW) as the identification is also needed to be simulated. In the meantime, several improvements have been implemented by National Instruments concerning the interfaces in LabVIEW which facilitates the integration. Yet when the work has been started, there have been no deterministic high-speed SHWFS available that offered direct support for the real-time LabVIEW. Moreover, the integration of Simulink models has been rather inefficient.

To overcome the mentioned limitations and to maximize the performance, a new rapid control prototyping (RCP) system has been drafted and developed which was briefly presented in the following Sec. 5.1 [69, 65]. The application of a commercial RCP system such as the systems from Digital Signal Processing And Control Engineering (dSPACE) GmbH was not viable since the costs would be immense. The proposed RCP approach is based on a self-development of an field programmable gate array (FPGA) peripheral component interconnect express (PCIe) card being installed into a Linux hard real-time capable PC. The initial cost in term of hardware for this RCP system is quite low whereas the required manpower for building such a system is demanding.

In the following, achieved results are presented in detail, concerning the procedure and the interaction of all components and algorithms. These give evidence to a superior performance compared to the former setup. It is fair to make clear at this point that it is also possible to accelerate LabVIEW code by using an FPGA card. However, the initial costs and complexity are much higher because LabVIEW and MATLAB are necessary. The adaptation and maintenance also is more involved than with the proposed RCP system.

² see <http://www.ni.com/tutorial/6477/en/>

5.1 System Setup

The reason for the self-development of a real-time computer with a completely new interface card based on PCIe has been briefly motivated in the introduction of this chapter.

On the one hand, the self-development has been considered as advantageous due to the desired performance of the AO system. There have been massive restrictions in the former LabVIEW based approach, resulting in computational, hard real-time and complexity limitations. On the other hand, another reason was that for successive application in the industry, the solution must be easy for use, being compact in size and highly adaptable. In a report from 2006 of the European Southern Observatory (ESO) about the European Extremely Large Telescope (E-ELT) [46] the necessity of using parallel approaches has already been addressed. It was highlighted that further improvement can only be significant when using parallel approaches because otherwise the complexity is likely to explode.

Bearing this information in mind, the application of an FPGA for (at least partially) preprocessing the SHWFS data, controlling and measuring the actuators is a necessity. From the position of a control engineer, the RCP must ideally be directly compatible with MATLAB in order to minimize the number of interfaces since MATLAB is the tool being used here for designing and simulating the controller. But at the same time, it must be highly customizable if it is required to achieve maximal performance.

Note that the application of a dSPACE system from dSPACE GmbH³ would entail enormous investments. For this specific project, the required investment was not available. Also note that the development of extension cards for a dSPACE system based on an FPGA is not easily achievable. Such a development would have been also necessary since the SHWFS should be evaluated in hardware as far as possible. Nonetheless, a lot of work, which e.g. is necessary using a bare Linux hard real-time system, has already been carried out employing dSPACE GmbH. Using a suitable commercially available system would lead to faster results, thus, the time required from concept state to real results is distinctively less. But due to the previous reasons such a system is until now not available.

Consequently the first test has been carried out to judge whether the evaluation of the SHWFS on an FPGA is beneficial by using a Xilinx Spartan-6 FPGA with a CameraLink interface [67]. After a successful proof of the concept, a PCIe FPGA card based on a Xilinx Kintex-7 has been developed. The concept of the inherent Linux hard real-time system and the associated FPGA card is presented in the following Sec. 5.1.1.

³ dSPACE GmbH, digital signal processing and control engineering GmbH, is a German company providing tools for developing, testing and calibrating e.g. electronic control units in the automotive, industrial automation and many more.

5.1.1 Hardware platform

The basic ingredient is the utilization of a high-performance computer based on a Quad-Core i7 central processing unit (CPU) from Intel which runs a hard real-time Linux system as OS. Such a system is state-of-the-art and relatively cheap. For minimizing the design effort during experimental tests, a RCP approach is required. It is capable of applying quick changes, test diverse control algorithms and iterate towards a further improvement of the overall performance. The interconnection between the RCP and the optical breadboard setup is shown in Fig. 5.2 using the build-in PCIe card. This PCIe card has to fulfill special requirements, i.e. being capable of the camera interconnection and additionally the SHWFS evaluation. For this purpose, the decision to develop a custom PCIe FPGA board serving as the basis of the platform for incorporating the specific aspects of AOs systems has been made.

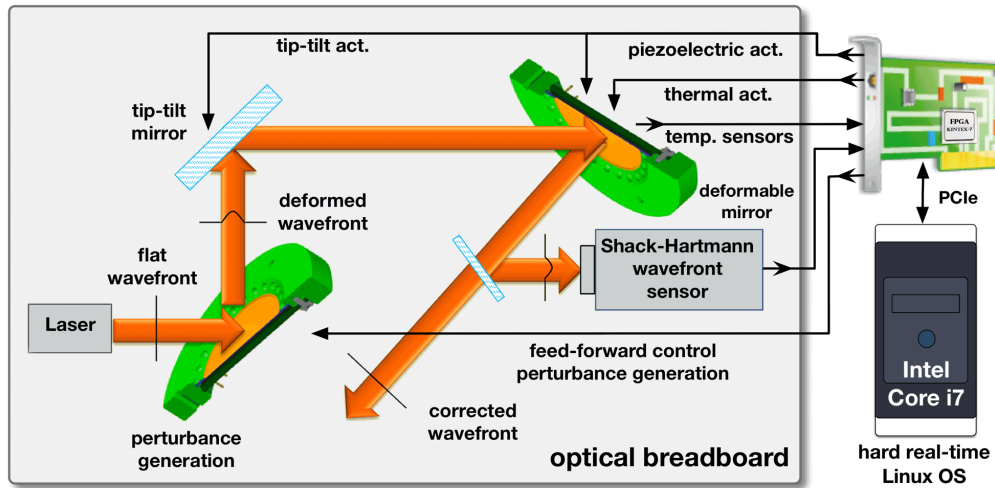


Figure 5.2: Overview over the new control loop concept [69]

The proposed concept replaces the LabVIEW solution [68, 66]. As far as it has been appropriate during the design phase, all available features of the PXI LabVIEW solution have been maintained or further improved. Therefore, the pinout of the very-high-density cable interconnect (VHDCI) connectors are compatible such that existing adapter and cables may be used from the old setup during integration of the new approach. The concept is either usable integrated within the performance computer or as stand-alone system, if e.g. power consumption and space is critical. For being independent of expensive and intricate hardware, except for the own-developed boards, the RCP resorts to real-time Linux as OS. The main purpose has been to create a system which has a better performance as the former PXI LabVIEW based system. The RCP has lower costs and will allow the entire control over any process of the system. The solution is specialized for high-performance control of AOs and shall not strive to replace existing high-performance (in general commercial) rapid prototyping solutions as for example from the dSPACE GmbH.

The fundamental basis of the concept is a Xilinx Kintex-7 FPGA. This FPGA

handles the communication to the host computer (if needed) via PCIe⁴ and interfaces the SHWFS (Imagine Optics HASOTM3 Fast) via CameraLink.

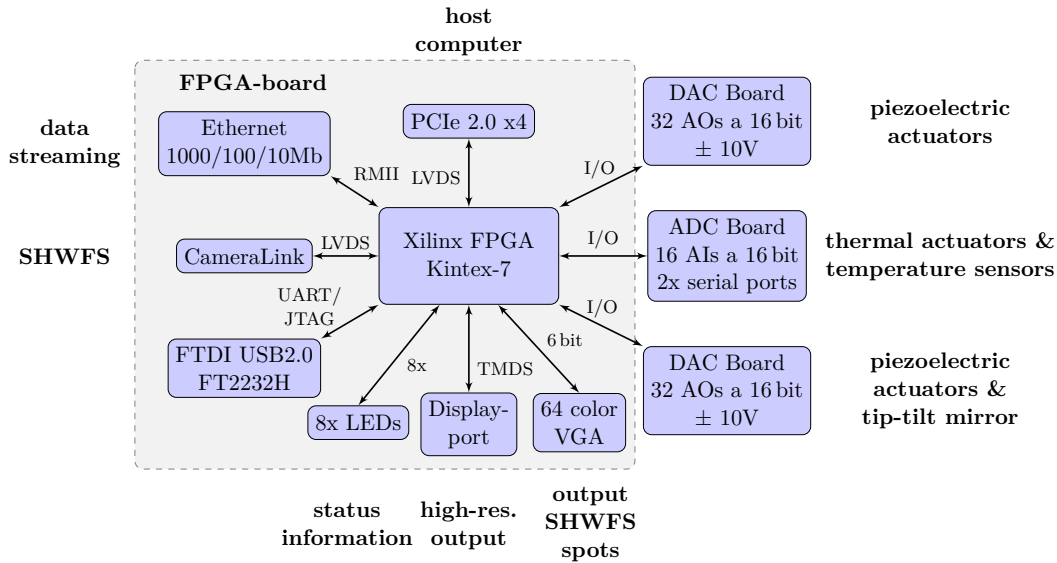


Figure 5.3: Diagram of the PCIe FPGA concept with the separate extension cards

The different interfaces that are implemented on the FPGA are listed in Fig. 5.3. On the right hand side, the extension boards are listed which drive a maximum of 64 piezoelectric or thermal actuators and analog inputs. These boards are directly connected via serial peripheral interface (SPI) to the FPGA for minimum delay, maximum performance and high flexibility. Fig. 5.4 combines the optical breadboard setup with the individual components and extension boards.

All analog signals shown in Fig. 5.4 are drawn in bold to be distinguishable from the digital signals. Each developed board is designed in the PCIe form factor such that the boards may be directly plugged into the computer housing for mounting the cards.

DM1 is used to generate specific disturbances in the AO setup. These disturbances are then compensated by a tip/tilt mirror and the DM2. As tip-tilt mirror the S-330.2SL from Physik Instrumente (PI) is employed. Each degree of freedom is controllable via a 0-10 V control signal. More information about the tip-tilt mirror is given in Sec. 4.2.1. DM2 has piezoelectric actuators and can additionally incorporate thermal sensors as well as integrated heaters [36, 37]. In general, DM2 is a specially designed deformable mirror to be implemented in high-power laser systems. The laser-induced mirror deformation can be compensated by controlled mirror heating as shown in [92]. Some mirrors offer mirror membrane buried functionalities, e.g. five thick-film heaters and nine thick-film temperature sensors. Notwithstanding, this dissertation focuses on ordinary piezoelectric actuators as there is no mirror so far, which incorporates all functionalities in one DM. Due to technical reasons and

⁴ PCIe, officially abbreviated as PCIe, is a high-speed serial computer expansion bus standard designed to replace the older PCI.

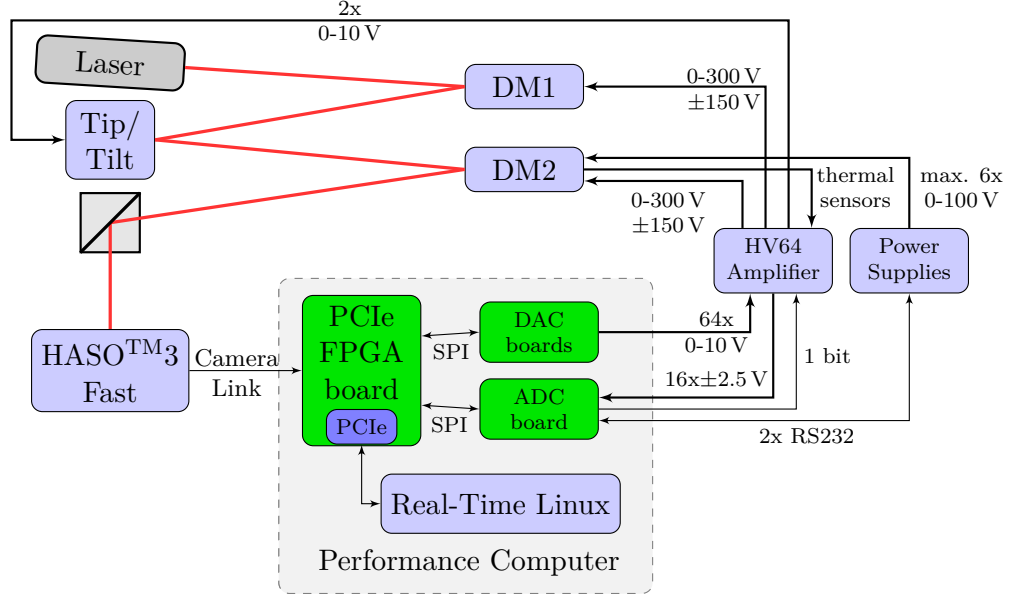


Figure 5.4: Overview over the complete concept [69] with all involved components

the corresponding time-line, older versions of the DM are used within this thesis.

The HV64 is an amplifier consisting of 64 channels for driving piezoelectric actuators. Each channel has a cut-off frequency (-3 dB) of approximately 50 kHz when a capacitive load of 6.6 nF is connected. Additionally, the HV64 incorporates a front-end for amplifying low current/voltage signals of the thermal sensors. The thermal sensors are amplified by the HV64. The signal is then captured by the analog-digital converter (ADC) board. To support up to 32 analog channels, a digital signal can be used to control the integrated analog multiplexer in the HV64. Therefore, with only 16 analog channels, 32 analog signals may be captured. The integrated heaters are controlled via external power supplies where each power supply consists of three buck-converters to compensate for the different temperatures of the DM. These power supplies have been designed in addition to the RCP, but are not part of this dissertation.

The performance computer receives the slope information which is being calculated based on the captured image of the HASO™3 Fast through the FPGA. With this information the control/error signals are computed as discussed in Sec. 4.2.3. Therefore, an FPGA board, two digital-analog converter (DAC) boards and one ADC board have been developed and integrated into the setup.

As an additional feature, the ground level of the outputs of the HV64 can be changed to +150 V such that positive as well as negative deflections are possible with the DM. This is advantageous compared to simply using an offset voltage of e.g. 150 V for all output channels since the creeping effect of the piezoelectric actuators will not occur when 0 V is applied. Thus, the HV64 behaves as a ± 150 V amplifier when the ground is externally supplied with +150 V.

For the presented experiments, the setup with ± 150 V is always used to suppress the

creeping while still having positive as well as negative deflection for surface shaping.

The actual real-time computer is based upon an Intel Core i7 4771 (Haswell architecture) mounted on an Asus H87-Pro main-board. As OS Ubuntu 14.04 LTS with Linux Kernel 3.10.18 is used which is patched with RTAI 4.0 to be real-time capable. For completeness, Fig. 5.5 visualizes the constructed PCIe FPGA card and the corresponding input/output cards which are installed in a medium-size computer housing.

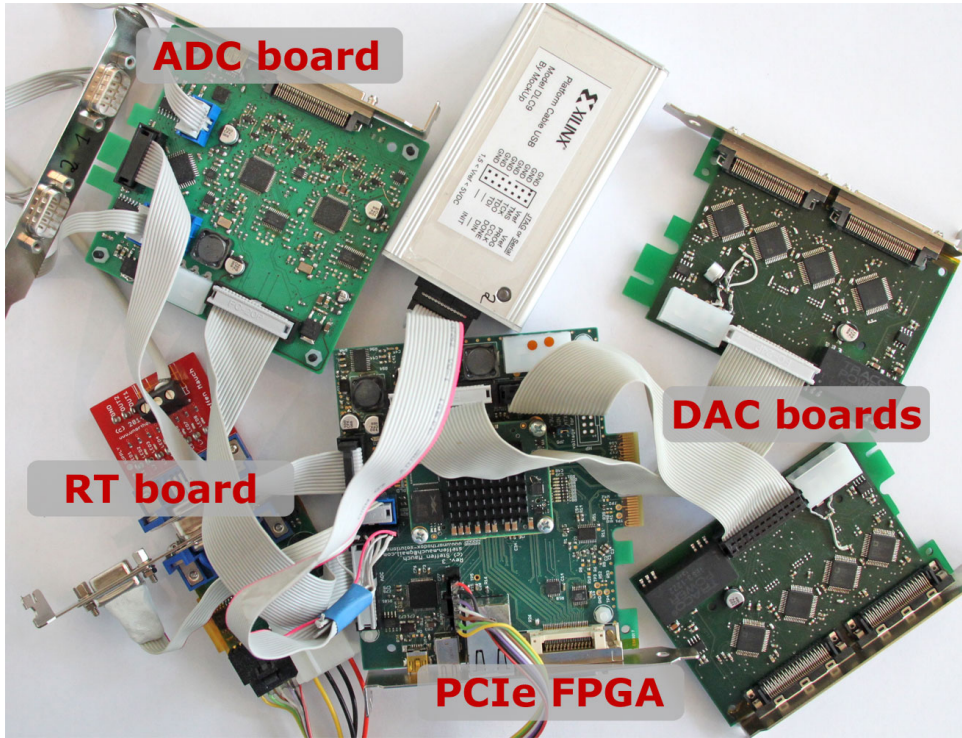


Figure 5.5: Developed PCIe FPGA card and the additional input/output cards

Fig. 5.6b shows the different cards that are mounted in the corresponding computer housing whereas Fig. 5.6a depicts the connectors.

5.1.2 Software platform

Real-time operating system

The selection of a suitable real-time operating system (RTOS) is crucial because it is the fundamental basis when setting-up a real-time system. Instead of Windows RT, vxWorks or any other available RTOSs, a specially patched Linux kernel has been chosen since such a Linux based system offers the possibility to analyze and modify everything as desired. The Linux kernel is an open-source project. It is very efficient regarding resource consumption and supports a variety of mainstream hardware.

For achieving (hard) real-time, e.g. having detailed information regarding run-time and latency, the real-time application interface (RTAI) 4.0 patch has been applied to the standard Linux kernel version 3.10.18. The mode of operation of RTAI is visualized in Fig. 5.7. Nowadays, RTAI is based on the adaptive domain environment

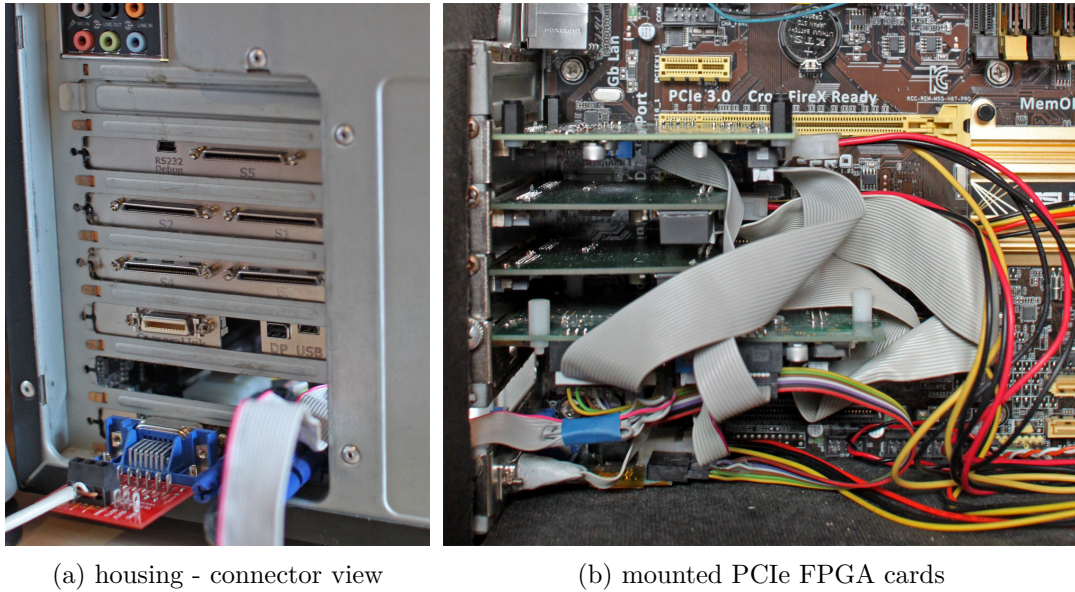


Figure 5.6: Installed PCIe FPGA cards in the housing

for operating systems (Adeos)⁵ kernel patch to achieve (hard) real-time behavior. The bypassing of the interrupts is carried out with the real-time hardware abstraction layer (RTAI). As long as the RTAI modules are not loaded, case A is active, as shown in Fig. 5.7. Therefore interrupts are handled in the same way as when using a non-patched standard Linux kernel. Although, if the RTAI modules are loaded, case B is active whereas the interrupts are assigned to the real-time extension. Nevertheless afterwards, the interrupts are also forwarded to the RTHAL.

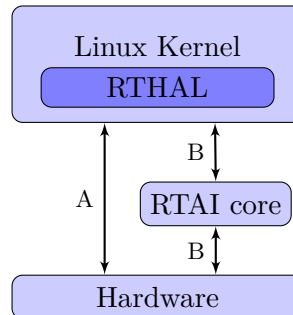


Figure 5.7: RTAI principle when RTAI core is active or inactive [69]

An enormous advantage of RTAI is the feature of user-space programs to be real-time capable by using LXRT. In contrast, the RT-Preempt patch only modifies the kernel to be fulfill real-time requirements and does not offer user-space real-time capabilities⁶. The possibility to use the user-space simplifies drastically testing because programming errors do not lead to e.g. a kernel panic. Additionally, libraries such as Eigen⁷ may easily be utilized. Therefore, considerably less effort has to be made for

⁵ The purpose of Adeos is to provide a flexible environment for sharing hardware resources among multiple operating systems, or among multiple instances of a single OS.

⁶ see https://rt.wiki.kernel.org/index.php/RT_PREEMPT_HOWTO

⁷ Eigen is a C++ template library for linear algebra. Eigen can calculate with scalars, matrices

exploiting the maximum performance of the CPU. When a kernel-space application can only be executed as real-time process instead of a user-space process, the inclusion of processor extensions such as AVX and SSE is much more complicated. So far no library is existing that can be utilized in a straight-forward manner. Therefore, intrinsic functions have to be used or even the special functionality has to be programmed in assembler, manually. Additionally, the utilization of floating point arithmetic in the kernel space is not as trivial as it is in the user-space since the registers are not automatically saved and restored.

Rapid control prototyping (RCP) implementation

RTAI already contains a target for the Simulink Coder. Using the target, a real-time executable can be generated from a Simulink model. As long as standard Simulink and RTAI/Linux Control and Measurement Device Interface (comedi) blocks are used, nothing has to be programmed or adapted. Generally, the blocks must be compatible with Simulink Coder such that C/C++ code can be generated automatically. RTAI also supports ScicosLab and Scilab which are open-source alternatives for the commercial application MATLAB. Thus the high-priced MATLAB license can be avoided for the RCP system if desired.

The involved layers are shown in Fig. 5.8 if the Simulink Coder is used to generate the code for RTAI. As RTAI has direct support for comedi, an abstraction layer for the measurement devices is available. Taking advantage of the abstraction layer a comedi driver has been developed for the PCIe FPGA card. Nowadays, this abstraction layer is directly shipped with the Linux Kernel but when being used by RTAI, comedi is specially patched to achieve better performance for supporting hard real-time.

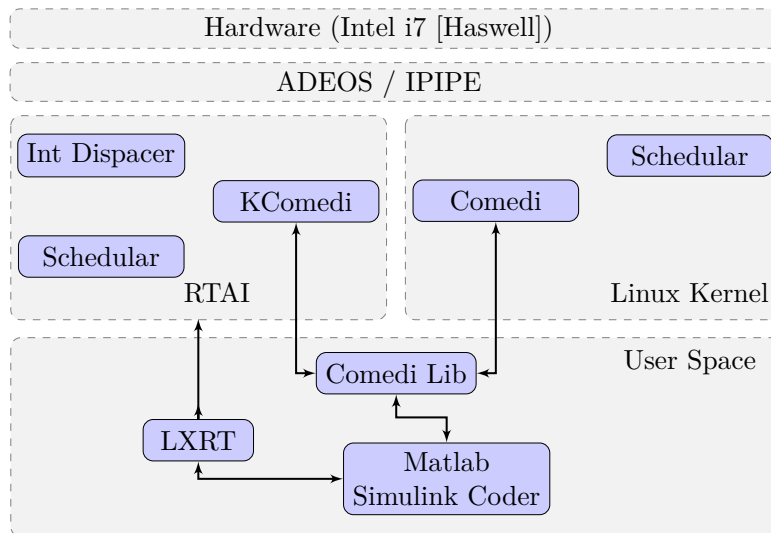


Figure 5.8: Block diagram of the different abstraction layers used in RTAI/LXRT

and vectors. It has numerical solvers and related algorithms already implemented with support for processor specific extensions such as advanced vector extensions (AVX) and streaming SIMD extensions (SSE).

In Fig. 5.9 the sequence of the different steps are shown when executing of the Simulink Coder. The 'rtmain.c' file contains the basic structure delivered by RTAI, e.g. the task handler, and maintains the application programming interface (API) of RTAI and comedi.

Currently an Intel Core i7 4771 quad core processor with Haswell architecture supporting the advanced vector extensions 2 (AVX2) and fused multiply-add instruction 3 (FMA3) instruction sets is used as CPU. These extensions to the x86 instruction set architecture for microprocessors from Intel and AMD are single instruction multiple data (SIMD) instructions which facilitate demanding arithmetic calculations such as matrix multiplications of large dimension by parallel computation of multiple multiplications in a single instruction cycle.

In MATLAB 2013b, Simulink Coder using GNU Compiler Collection (GCC) as a compiler is not capable of accelerating matrix calculations or state-space models. Therefore, special s-functions have been written in C/C++ which are using, instead of the built-in functions, the special processor features such as AVX2 and FMA3 [65]. In Appendix A.6 some results regarding performance are illustrated including matrix multiplication, state-space models and additionally, calculating Zernike polynomials based on SHWFS slopes.

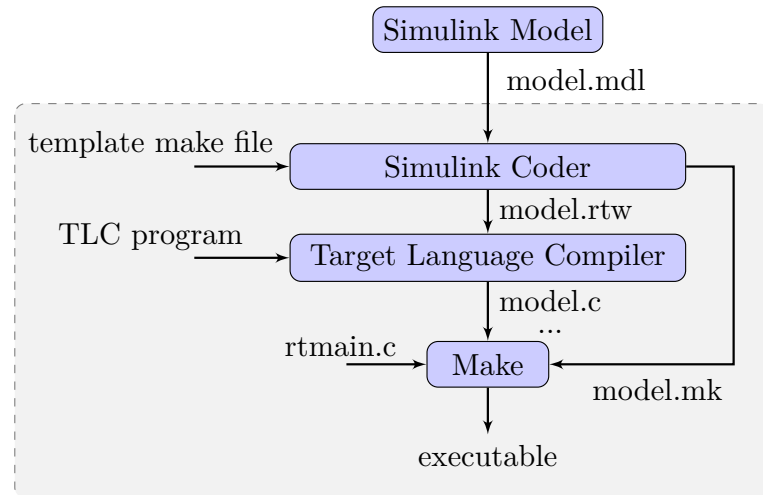


Figure 5.9: Invoked modules when an executable for RTAI is built in MATLAB using a Simulink model [69]

Interfacing the PCIe FPGA card via comedi

For the PCIe FPGA card a kernel driver based on comedi example skeleton driver has been developed. The handling of physical units, differentiation of analog and digital signals is already accomplished in comedi. The interface for accessing the functions coded in the kernel-space from the user-space are already implemented as well. Therefore, Simulink Coder can be used with LXRT to run RTAI real-time tasks in user-space without much effort with own developed hardware.

For accessing the hardware in Simulink using RTAI/comedi, several s-functions in the

RTAI Simulink library are available for e.g. writing analog values. Thus, the basic ability is given and may be extended to the other required features such as memory reading. As an appreciated side-effect when using comedi, the functionality of the PCIe FPGA may be used with python since a python binding for comedi and RTAI is given. Recording the AIF with Simulink is not comparably comfortable as with a python based implementation as python can e.g. save the measured values easily as structured and preprocessed ‘.mat’ file. This file may then be edited further in MATLAB. Furthermore, sequential program flow, as required for capturing the AIF, may much simpler be realized in python than within a Simulink model. Notwithstanding, capturing the AIF can be implemented in Simulink using the stateflow toolbox.

Fig. 5.10 shows the basic structure of the comedi device driver which is being grouped into four sub-devices. By accessing sub-device three, it is also possible to trig-

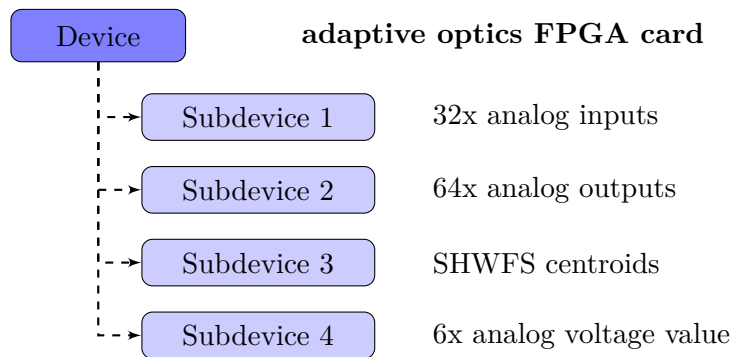


Figure 5.10: Overview over the implemented comedi device driver [69]

ger the SHWFS from a RTAI application such that the frame-rate is synchronous to the real-time system [65] using the available CameraLink trigger input of the SHWFS. Being synchronous to the control task is a highly desired feature.

Additionally, the driver creates a special entry in the ‘/proc’ directory such that e.g. the image of the SHWFS can be dumped easily using linux command line tools. Furthermore several configurations regarding the SHWFS and other functionality can be changed easily by writing to this entry. The centroids may be dumped based on this method as well. Therefore, these dumps can be used for validation of the implemented algorithm itself and to test new approach with MATLAB.

5.2 Experimental Results

For comparing the applicability of the \mathcal{H}_∞ approach, especially the non-smooth μ -synthesis, several experiments have to be conducted. The results of the experiments have been harmonized with previously realized simulations. Hence, the simulation model has been matched with the experimental setup as far as possible to remove any discrepancy.

As previously obtained results have been gathered with the tip-tilt mirror and the former setup [68, 66], the identification of the improvement between the new proposed RCP system and the former setup is possible. Therefore, novel experiments have been performed with the tip-tilt mirror. In this chapter, the simulation and experimental results are presented individually for the tip-tilt and DM application.

The utilized Simulink Model for code generation is depicted in the Appendix (see especially Fig. A.7). It is important to note that the outputs of the SHWFS block are slopes for each sub-aperture. The calculated values using the control matrix Γ are called in the following error values having an arbitrary unit (a.u.) as these values in fact are the desired wavefront subtracted by the actual wavefront. The error value does not have a SI unit itself. The desired wavefront is assumed to be perfectly flat, thus, the error value is solely the output after the calculation with the control matrix. This error value correlates with the individual required deflection of each actuator. For calculating the slopes a reference wavefront is applied. Therefore before running the script to calculate the AIF, the wavefront is captured and used as reference wavefront. Thus, the reconstructed one does not necessarily represent the actual wavefront as the reconstructed wavefront is based on a relative measurement.

5.2.1 Tip-tilt control

When considering the AO system being reduced to only the tip-tilt control, the main difference is that the developed RCP system introduces less delay due to the evaluation of the SHWFS. The tip-tilt mirror itself remains unchanged compared with the previously LabVIEW based setup. Thus the model of the tip-tilt mirror is the same. Especially the time-delay of the SHWFS within the LabVIEW based setup is unknown [66]. The delay was uncertain in the range of 3 up to 4 ms. With the current setup, the time-delay is $1050\mu s$ while the exposure time has been set to $100\mu s$. Please note, the time-delay is always a multiple of the controller sampling rate (see Sec. 4.2.3).

The primary published experimental results being obtained with the tip-tilt mirror are a good basis to compare with [68, 66] to numeralise the enhancement of the setup. These results are based on the former LabVIEW based concept; for more information the reader is referred to the introduction in Chap. 5.

Simulations with the tip-tilt mirror

The controller sampling rate has been chosen to 1600 Hz. As the SHWFS is not capable of delivering such a high frame-rate, the SHWFS is triggered with halve frequency. Therefore, the resulting delay is equal to $\frac{1}{800}$ s. Only considering the time-delay of the SHWFS, the frame-rate may be increased without having more than one frame delay (see Sec. 4.2.3). Keeping in mind the delay of the RCP and the time necessary for the direct memory access (DMA) transfer (see Appendix A.6.5), the SHWFS sampling time has been set to $\frac{1}{800}$ s.

Note that the exposure time has been adjusted to only $40 - 50 \mu\text{s}$ when comparing with the values given in Appendix A.6.5. This lower exposure time is sufficient when the SHWFS is illuminated well. Therefore, the SHWFS rate may be increased up to $850 - 860$ Hz while still having only one frame delay.

Nevertheless some margin regarding the resulting time-delay should be added. This small reserve does not degrade the performance remarkably and a slight increase of the exposure time should not result in the design of a new controller. Such an adaption would e.g. be required due to changes of the incident laser intensity. This could be a result of a different disturbance mirror with lower reflectance or in an application where less incident light can be used for wavefront sensing.

The controller for the tip-tilt mirror has been synthesized with the non-smooth μ -synthesis, the basic procedure being described in Sec. 4.1.5. To represent the uncertainty and un-modeled high frequency components, the tip-tilt model $G(s)$, see Equ. 4.50, has been combined with additive uncertainty. The resulting model for the tip-tilt mirror is

$$G_{\text{tip/tilt-Total}}(s) = \begin{pmatrix} G_{\text{tip/tilt}}(s) & 0 \\ 0 & G_{\text{tip/tilt}}(s) \end{pmatrix} + \begin{pmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{pmatrix} / 10 \quad (5.1)$$

where $\Delta_{1/2} \in \mathbb{C}(s)$ and $\|\Delta_{1/2}\|_\infty \leq 1$.

As weighting matrices the transfer functions

$$W_o(s) = 1000 \frac{(10s + 1)(s/1200 + 1)}{(100s + 1)(s/6 + 1)} \quad (5.2)$$

$$W_u(s) = 0.3 \frac{(s/80 + 1)}{(s/10000 + 1)} \quad (5.3)$$

have been used. The transfer functions $W_o(s)$ and $W_u(s)$ are used for shaping the output and actuating value during synthesis (see Fig. 4.7). $W_d(s)$ is set to identity, thus, no specific shaped disturbance is assumed. The basic functionality of low- and high-pass filters and of each weighting function is explained in Sec. 4.2.6.

With the non-smooth μ -synthesis, an \mathcal{H}_∞ controller has been synthesized by using the weighting matrices. For an objective comparison of the results with those in [66], the order of the controller has been set to eight. Fig. 5.11 depicts the bode diagram of the synthesized controller for the tip case. The tilt case is almost identical, thus not

visualized. Comparing the cut-off frequency with the formerly achieved frequency,

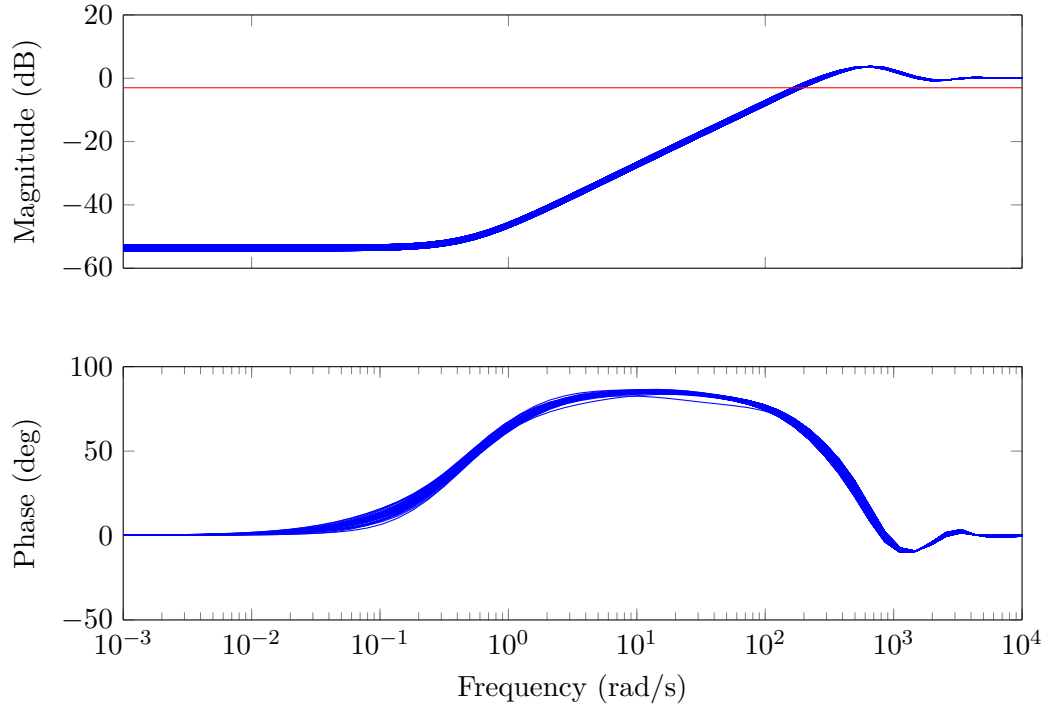


Figure 5.11: Simulated bode diagram with uncertainty (transfer function for tilt disturbance to tilt actuator); red line marks the -3 dB line (intersection is the cut-off frequency)

based on the former experimental setup, an improvement of more than factor of two is observed. The cut-off frequency is increased to approximately 240 rad/s. This improvement is caused by the smaller time-delay which additionally now being constant and deterministic.

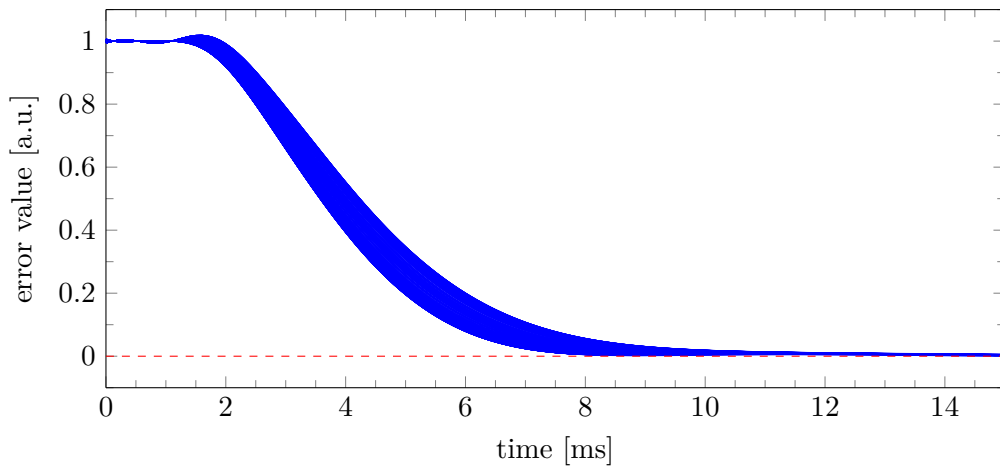


Figure 5.12: Simulated step response with uncertainty with Padé-approximation (transfer function for tilt disturbance to tilt actuator)

The step response of the closed-loop system is given in Fig. 5.12. The time required to compensate for a step disturbance is approximately 7 to 8 ms whereas the

LabVIEW based system has shown a time between 12 to 16 ms. Both simulations (Fig. 5.11 and 5.12) do not take noise into account. Furthermore, for incorporating the delay the Padé-approximation has been utilized. Any introduced error due to the replacement of the time-delay through the Padé-approximation would not be apparent so far, as the approximation is used for synthesis and simulation.

Therefore, simulations have been conducted with noise and the un-approximated time-delay. The result, based on a Simulink model, is visible in Fig. 5.13. This Simulink model does replicate the real circumstances and thus, the experimental setup as far as it is possible and suitable. Furthermore, this simulation also reproduces the differentiation of the wavefront as well as the influence of the reconstruction using the control matrix. The employed MATLAB scripts as well as Simulink models have been published (see Appendix A.10). Likewise, the synthesized controller is published as ‘mat’ workspace; thus, the coefficients of the controller are not shown here.

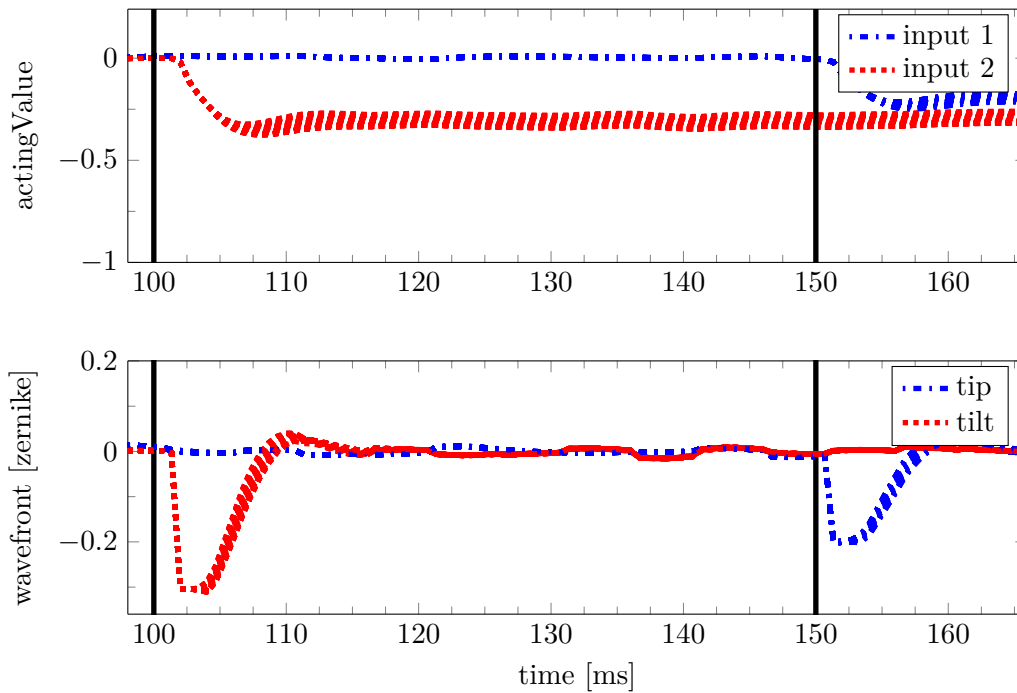


Figure 5.13: Simulated step response with uncertainty, noise and no approximation for time-delay (at 100 ms tilt disturbance; at 150 ms tip disturbance)

It is worthwhile to classify the result in terms of the step response of the tip-tilt mirror which is given in Fig. 4.10. Analyzing this response independently, it can be observed that the employed tip-tilt mirror itself does not have a fast step response compared to the time-scale of the closed-loop. Therefore, the performance cannot be increased further. The tip-tilt mirror itself contains an inner closed-loop relying on the strain gauges for controlling the deflection. The step-response of the tip-tilt mirror exhibits a distinctive damped behavior. Tuning or replacing the inner closed-loop of the tip-tilt mirror may improve the step response significantly. Thus, actually the tip-tilt mirror limits the performance.

However, in the time interval of 6.25 ms only five images of the wavefront, i.e. error values, are available for the control adjustment. Thus, in order to improve the performance further, a faster SHWFS is required that has a larger frame-rate while maintaining the time-delay and, additionally, a faster tip-tilt mirror.

Experiments with the tip-tilt mirror

For the validation of the simulation results, several experiments have been performed. Based on these results, the simulation model is validated such that the plant is modeled correctly and the gathered simulation results may be questioned.

Hence, the step response of the closed-loop has been recorded. This step response is given in Fig. 5.14. Based on the data, it is apparent that the controller is able to compensate for the step disturbances with frequency of 2 Hz. This is the expected behavior and the closed-loop is stable. At time instance $t = 5$ s, a positive step in tip/tilt has been applied. As the step has a duration of $t = 0.5$ s, for the next half second, no disturbance is applied. The positive as well as negative spikes are the result from the rising and falling edge of the step disturbance. Based on Fig. 5.14

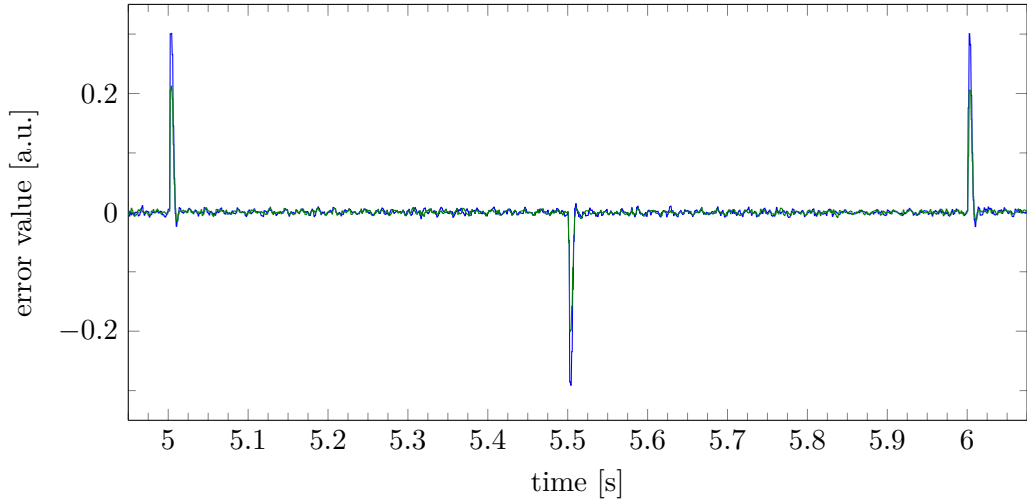


Figure 5.14: Step disturbance of 2 Hz with enabled tip-tilt controller; $f_{\text{controller}} = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz

it is hard to judge whether the performance is different from or identical with the simulation. To this end, Fig. 5.15 is based on the same data as Fig. 5.14, but the x-scale has been zoomed. Hence, the time required to reject the step disturbance can be determined to 7-8 ms. This is almost identical to the simulation and proves the modeling and simulation, whereas the model of the tip-tilt mirror has already been validated in [66].

The results recorded for random values and/or sinusoidal signals, generated as disturbances with the DM, are dropped here since the main focus of this subsection is not on tip/tilt control. Instead, such experimental data are presented for the DM case.

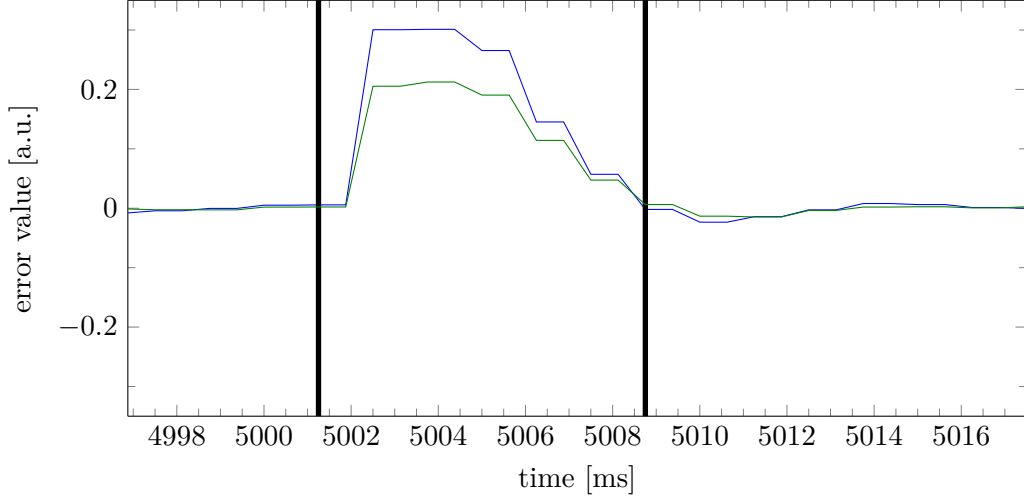


Figure 5.15: Step disturbance of 2 Hz with enabled tip-tilt controller (zoomed in for comparison with simulation characteristic); $f_{\text{controller}} = 1600 \text{ Hz}$ and $f_{\text{shwfs}} = 800 \text{ Hz}$

5.2.2 Deformable mirror control

In this subsection, simulation and experimental results for the utilized DM, constructed and assembled by the Fraunhofer IOF (see Sec. 2.1 and 4.2.2), are presented. The overall model for the deformable mirror having 40 actuators (DM40) (see the identification and modeling in Sec. 4.2.2) has many states, thus, is a large dimensional model. Using transfer functions, the resulting transfer matrix for the DM40 consisting of 40 actuators is of dimension 40×40 . Due to the fact that in the optical setup the laser beam diameter is shaped such that only the inner 24 actuators are illuminated, the dimension reduces to 24×24 . Each entry of the transfer matrix is of third order. The outer 16 actuators cannot be observed without additional effort since the influence of these are only indirectly measurable. In Sec. 4.2.3 a non-linear reconstruction algorithm taking actuator saturation into account has been presented. This approach is able to cope with ambiguity, but the algorithm restricts the possible frame-rate of the SHWFS. As the focus is put on performance, the standard approach with the pseudo-inverse has been applied and the other 16 actuators have thus been neglected.

To demonstrate the performance of the closed-loop, several selected signals such as sinusoidal and step functions have been used to steer the actuators of the disturbance mirror. The disturbance mirror is an additional DM (see Fig. 5.2) for generating reproducible disturbances for also reproducible controller assessment. Experimental results of the closed-loop system are compared with the simulation results and elaborately discussed. The benefit of a setup consisting of two DMs (having a specific disturbance mirror additional to the standard DM) is that instead of pseudo disturbances by signal injection, the behavior having real spatial disturbances can be analyzed.

The primary goal is not solely to achieve good performance regarding speed and accu-

racy, but also to assess that the identified model of the individual components matches well with the given assignment. Therefore the experimental results are compared with the expectation gathered from the simulational results.

As for the tip-tilt case, the employed MATLAB scripts and the different identified models, e.g. for the DM40 are available in App. A.10. The synthesis of the \mathcal{H}_∞ controller using the non-smooth μ -synthesis (see Sec. 4.1.5) initialize the parameters with random values if the parameters are not explicitly specified. Thus, the resulting controller may differ even when using the same models and weighting functions for controller synthesis by default.

The initial step of determining a stabilizing controller may turn out to be quite costly in terms of time. Under some circumstances, it may be beneficial to use the classical \mathcal{H}_∞ synthesis to retrieve a full-state controller first. By application of a reduction/truncation method the full-state controller may then be adjusted to the desired order or structure which is desired for the non-smooth μ -synthesis. As an initial guess for the structured controller, the calculated values are used to initialize the parameters.

Simulations for DM40

Previous to the simulations, the controller which stabilizes the plant has been synthesized based on the non-smooth μ -synthesis approach. The result is a structured controller showing a fixed-order. As a PI-controller can also be written in state-space form, the controller can even be a multiple input multiple output (MIMO) PI-controller. The order of the controller is almost arbitrary. In the past, typically, PI-controllers have been used for controlling each actuator individually. Thus for 24 actuators the number of states results to 48 in total. Using individual controllers for each actuator does not incorporate the present coupling. Hence, statements regarding stability are difficult to obtain. For MIMO systems which include time-delay, there is no procedure yet to design the controller without using empirical methods. The problem gets worse if the controller is required to be robust against specific uncertainties since these extended models must also be incorporated within the design of the controller. Otherwise, the controller can only be tested if it is stabilizing even when the system is uncertain by e.g. using the μ analysis. Notwithstanding, using the same number of states, in this case 48 states for a state-space controller, is a good start for a MIMO controller.

The controller for the DM has been synthesized according to the procedure in Sec. 5.2.1 by applying the non-smooth μ -synthesis (see Sec. 4.1.5). The uncertainty due to the model error of the DM, e.g. the hysteresis being approximated as time-invariant and the un-modeled high frequency components, is represented as uncertainty. The resulting model for the DM reads

$$G_{\text{DMtotal}}(s) = G_{\text{DM}}(s) (\mathcal{I}_{24} + \Delta_{\text{DM}}/5) \quad (5.4)$$

with $\Delta_{\text{DM}} \in \mathbb{C}(s)^{24 \times 24}$ and $\|\Delta_{\text{DM}}\|_{\infty} \leq 1$.

The bode plot of the closed-loop is shaped analogously to the procedure for the tip-tilt mirror. As weighting matrices the following transfer functions are used

$$W_o(s) = 2000 \frac{(10s + 1)(s/1200 + 1)}{(100s + 1)(s/6 + 1)} \quad (5.5)$$

$$W_u(s) = 0.4 \frac{(s/200 + 1)}{(s/10000 + 1)} \quad (5.6)$$

with more details in Sec. 4.2.6. $W_o(s)$ and $W_u(s)$ weight the output and actuator value (see Fig. 4.7).

The bode plot from input one to output one of the closed-loop system with synthesized controller $K(s)$ is shown in Fig. 5.16 without the weighting matrices. Several runs have been performed with different realizations for the uncertainty such that a set of curves is gathered. For obtaining the bode plot, the time delay has been approximated with the Padé-approximation which is also used for creating the model that is required for the synthesis step. The result from the synthesis step is a state-space controller which has 48 states with 24 inputs, 24 outputs and thus, dimension 48. The transfer function from input one to output one is exemplarily plotted in Fig. 5.16.

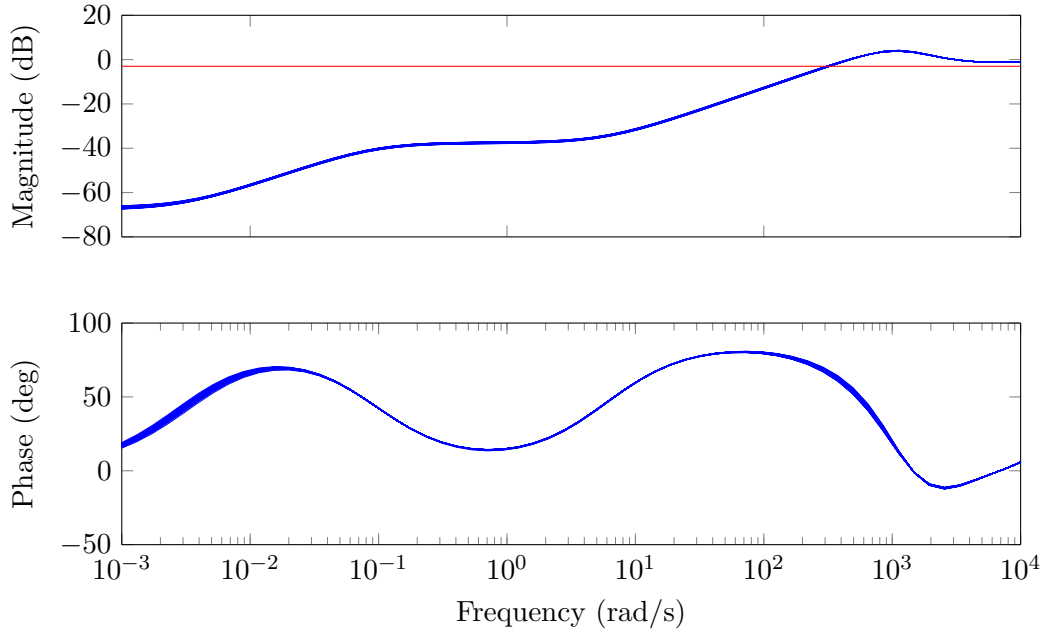


Figure 5.16: Simulated bode plot with uncertainty (transfer function for disturbance one to actuator channel one); red line marks the -3 dB line

The red line in Fig. 5.16 represents the -3 dB line. Thus, the angular cut-off frequency may be read off as $\approx 314 \text{ rad/s}$. Consequently, disturbances with a frequency up to $\approx 49.9 \text{ Hz}$ are partially suppressed. This means that the amplitude of a sinusoidal signal being contained in the output is only $\frac{1}{\sqrt{2}}$. When regarding the power instead of the amplitude, only half of the power regarding this specific frequency is left in the signal.

A small frequency band which starts from ≈ 400 rad/s to ≈ 2000 rad/s amplifies the input disturbance slightly. This behavior is equal to the tip-tilt closed-loop behavior. As the frequency tends to infinity, the amplification tends to one, thus 0 dB (see Fig. 5.16).

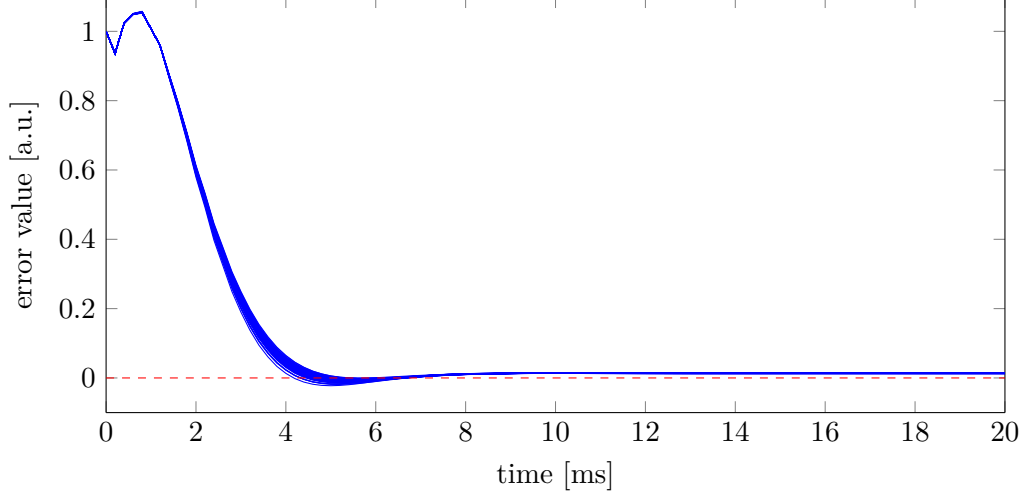


Figure 5.17: Simulated step response with uncertainty (transfer function for disturbance one to actuator channel one)

Fig. 5.17 depicts the step response for the same constellation as in Fig. 5.16. The required time for disturbance compensation is approximately 5 to 6 ms. It is also obvious that a steady-state error remains despite the large damping of more than 60 dB at 10^{-3} rad/s. This is not surprising as neither the plant nor the controller has an integral behavior. When regarding the first millisecond, the signal looks rather rippled. This behavior is an artifact of the Padé-Approximation and is visible just in the time-domain and not in the frequency domain.

In contrast to Fig. 5.17, Fig. 5.18 has been simulated by application of a Simulink model with similar noise as has been recorded during the experiments. Furthermore, the time-delay is a pure time-delay without using any approximation. The controller has been discretized with sampling rate of the real-time system and the discrete behavior of the SHWFS is considered combined with the application of the control matrix for actuator signal reconstruction as well. By comparison of Fig. 5.17 and 5.18 the match can be seen as rather good. Only the overshoot is slightly increased which vanishes when a perfect control matrix is used instead of the control matrix being gathered while subject to noise.

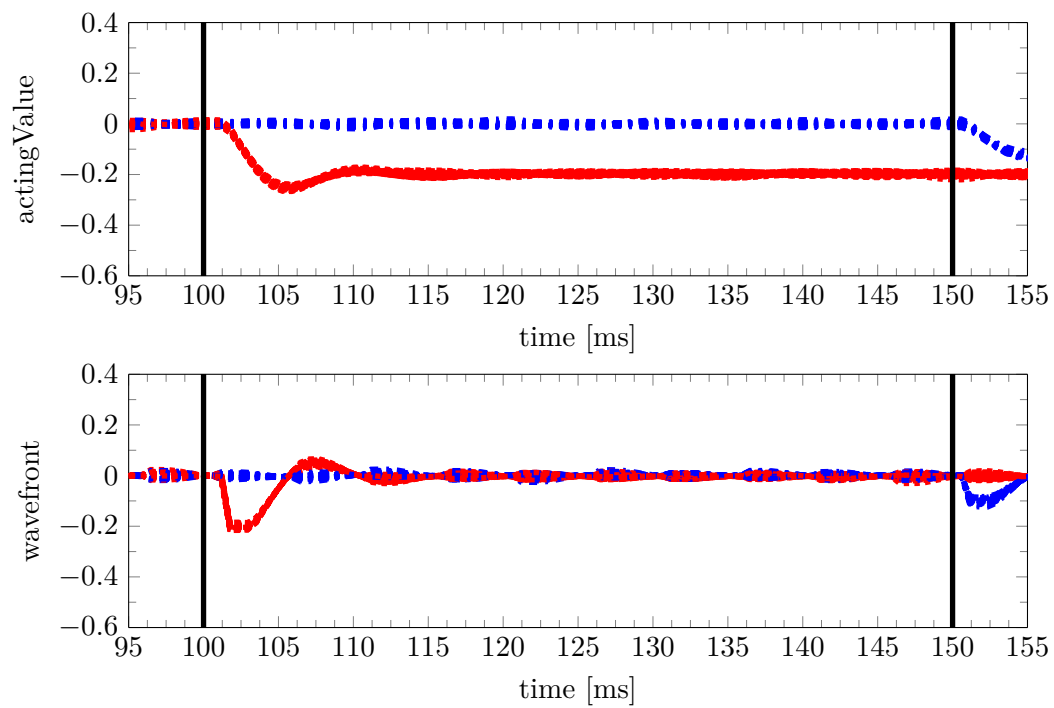


Figure 5.18: Simulated step response with uncertainty, noise and no approximation for time-delay (only first two actuators are shown due to clearness)

Experiments with DM40

During the experiments, by default the SHWFS slopes, a timestamp of the real-time system and additionally the signal written to the analog outputs are recorded. Hence, for the case of the DM40, 196 times two slope values, one timestamp and 24 analog outputs are stored for each time stamp of the real-time system.

Technically, 40 analog voltage outputs are written because the outer 24 actuators are biased to +150 V to have zero deflection. The HV64 shows in total a range of 300 V, but this range is split into a positive and negative voltage to be able to deflect in both directions. Therefore, only voltages of ± 150 V are possible. As the ground potential is set to +150 V effective ± 150 V are given.

The applied DM40 has a maximum peak deflection of only ≤ 300 nm at the maximum voltage of 150 V. Therefore, an error value of 0.1 corresponds to ≈ 30 nm deflection. The utilized SHWFS has a variance of $\approx 25 - 35$ nm peak to valley when ambient light is apparent and the SHWFS is not actively cooled (thermal noise of the sensor). This value has been identified by capturing long time-series and subsequent calculation of the variance based on the reconstructed wavefront with zonal or modal reconstruction methods. Due to this circumstance, the error value will never be equal to zero even when a real integral controller is applied as the measurement accuracy is not perfect.

During the experiments with the DM40 it became obvious that the standard mode when capturing the AIF was slightly worse than the hadamard method. The differentiation between the standard and the hadamard method is treated in Sec. 4.2.3. The better performance of the hadamard method is caused by the fact that the stroke of the actuators is rather low compared to the variance of the SHWFS itself. The accuracy of the AIF is crucial as this has an enormous impact on how well the wavefront is compensated.

For visualization purpose, either the wavefront is reconstructed based on the slope values or the calculated actuator values (called error value) are used. The calculation of the error values is described in Sec. 4.2.3.

These error values are shown e.g. in Fig. 5.19. At first the AIF has been captured and the control matrix has been determined based on the AIF. This calculation was done offline. For being able to record the error values, the slopes are multiplied with the control matrix which results in the error values.

As disturbance, a rectangular signal with 2 Hz frequency has been applied on one actuator of the disturbance DM periodically. Since the AIF has been taken with the maximum voltage possible, the error value in Fig. 5.19 has a permissible range of ± 1 . The range of the values should never exceed ± 1 because otherwise a deflection being greater than the maximum possible stroke of the corresponding actuator would be required, which is not achievable. This relationship is only given for the open-loop operation. During closed-loop operation, the error value is fed into a pseudo integral controller which evokes that the error value tends to zero. In this case, the output

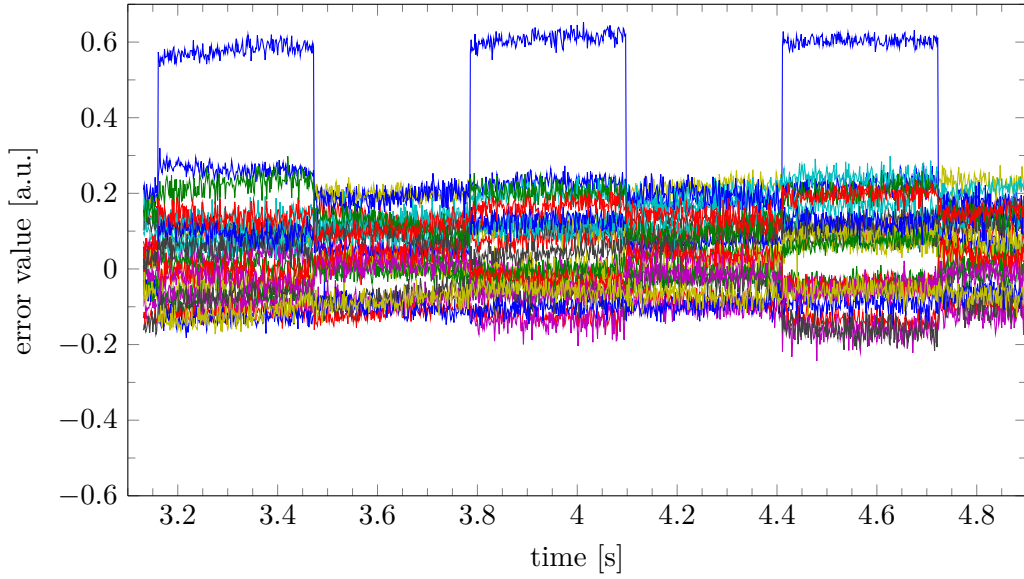


Figure 5.19: Visualization of the error value if applying a 2 Hz step disturbance and the controller is not activated (open-loop); sampling rate is $f_{\text{shwfs}} = 800 \text{ Hz}$

voltage has to be analyzed to detect saturation.

Examining the upper blue rectangular curve in Fig. 5.19, a slow increase of the error value can be seen at time instance 3.2 s. This corresponds to the creep behavior of the piezo actuator of the disturbance mirror. The disturbance mirror has been activated at time instance 1 s. This effect has not been modeled into the identified model of the DM but in any case, the controller must be capable of handling such a slow variation since the time-scale difference is immense.

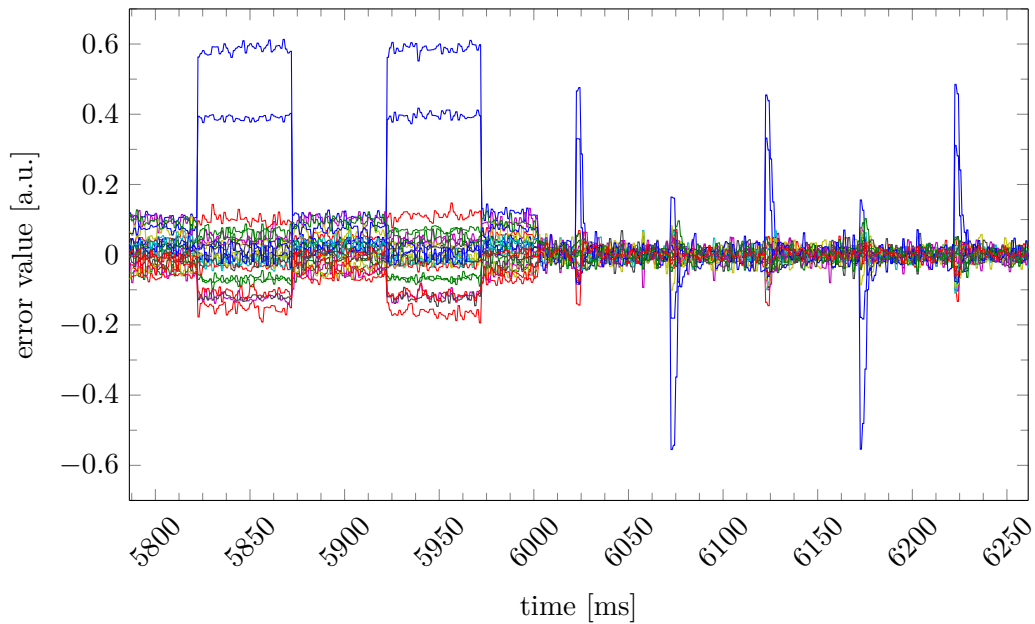


Figure 5.20: Application of a 10 Hz step disturbance while controller is switched on at $t = 6 \text{ s}$; $f_{\text{controller}} = 1600 \text{ Hz}$ and $f_{\text{shwfs}} = 800 \text{ Hz}$

To some extent, the measurement inaccuracy of the SHWFS explains the noisy signals in Fig. 5.19. Only one actuator of the disturbance mirror has been driven with a square signal, however, multiple signals show a square waveform in Fig. 5.19. This is caused by the different actuator layout of the disturbance mirror and the DM40. Actually, the number of actuators and actuator patterns of both mirror are not even the same. This detail is not important for acting as a disturbance mirror. However, crucial is the velocity of the disturbance mirror. As a result of Fig. 5.19 it is clearly obvious that the speed of the disturbance mirror is sufficiently fast. Applying a step leads to a step for the SHWFS as well while the edge is not slurred.

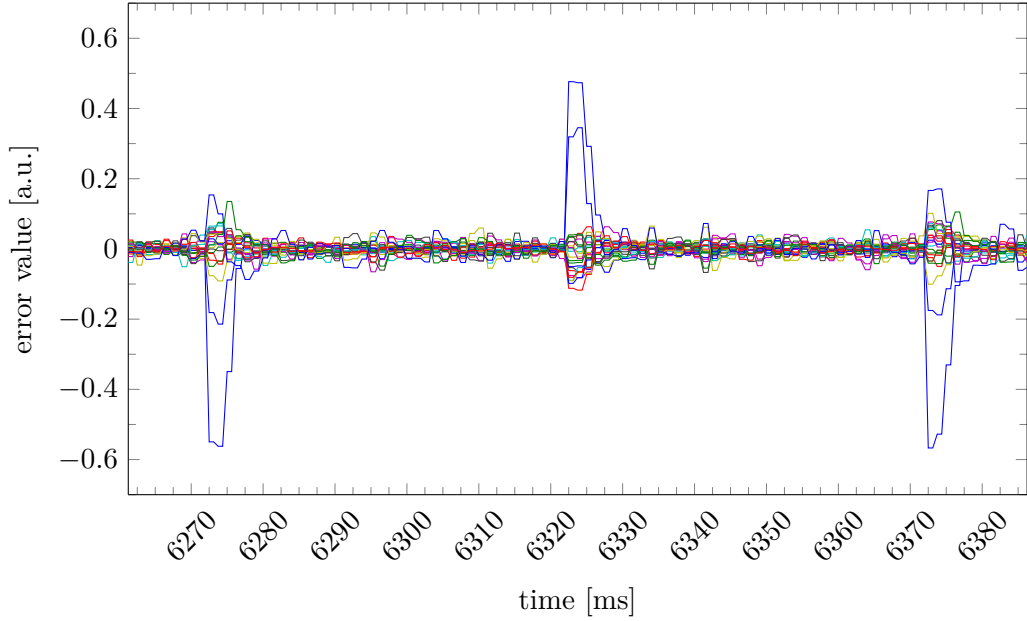


Figure 5.21: Application of a 10 Hz step disturbance with enabled controller; $f_{\text{controller}} = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz

The closed-loop result is depicted in Fig. 5.20. In this case, the controller has been switched on at time instance $t = 6$ s. Thus, after $t = 6$ s closed-loop operation is performed and the DM40 compensates the disturbances efficiently. It is clear that after $\approx 6-7$ ms the disturbance is almost completely rejected. To clarify this behavior in more detail, Fig. 5.22 is zoomed to Fig. 5.21, i.e. the time instance ≥ 6314 ms. Both figures are showing the control behavior in detail. Likewise noticeable, the noise band is rigorously reduced when compared with the open-loop measurements in Fig. 5.19.

Carefully analyzing Fig. 5.22, the error value is seen to be identical for the next sampling time. This may easily be explained since the controller sampling time has been chosen to twice the sampling time of the SHWFS. Higher sampling frequency of the controller is beneficial as the controller is a discretized version of the synthesized continuous one.

To emphasize the principle of operation and the effectiveness of the controller when rejecting the disturbance, Fig. 5.23 is to show the reconstructed wavefront for the given error values in Fig. 5.22. For clarity, every second wavefront has been reconstructed by

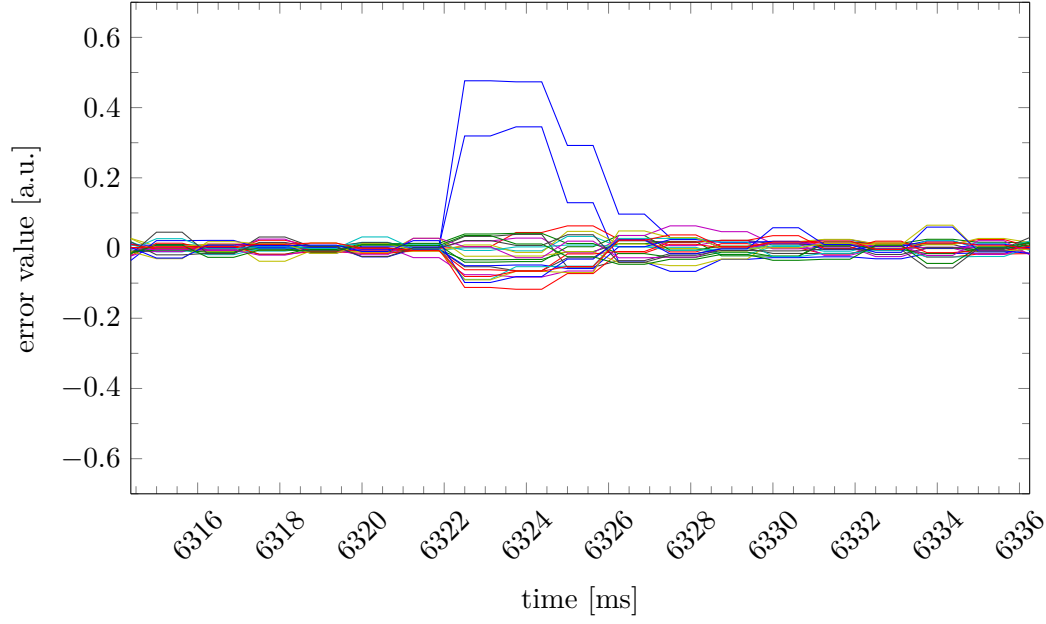


Figure 5.22: Same experimental data as in Fig. 5.21 but with zoomed x-axis to highlight the control behavior after a step disturbance had occurred, $f_{\text{controller}} = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz

applying the zonal reconstruction and then been plotted in Fig. 5.23. At time instance $t = 6321.88$ ms the wavefront is almost flat resulting in a peak-to-valley wavefront error of only 29.40 nm. Then an disturbance is present, thus, the peak-to-valley wavefront error is 293.19 nm, 307.33 nm, 158.55 nm, 44.80 nm and at $t = 6328.13$ ms only 45.63 nm. Thus, after five to six captured wavefronts, the peak-to-valley error of the wavefront is the same as before the disturbance has occurred. At time instance 6330.63 ms the peak-to-valley value returns to 35.48 nm. In the literature, the Strehl ratio, i.e. the peak-to-valley value or the root-mean square (RMS) error is used to judge the quality of an optical setup. It depends on the application which is more suitable as each method as its own advantages and disadvantages. The Strehl value, before and after the disturbance has occurred, is 1 and 0.99. A Strehl value of approximately one does not imply that the wavefront is flat as the SHWFS here is using a reference wavefront to calculate the slopes. In comparison, the reconstructed wavefront is almost flat in Fig. 5.23 as the reconstruction of the wavefront already uses the slopes which are based on the reference wavefront.

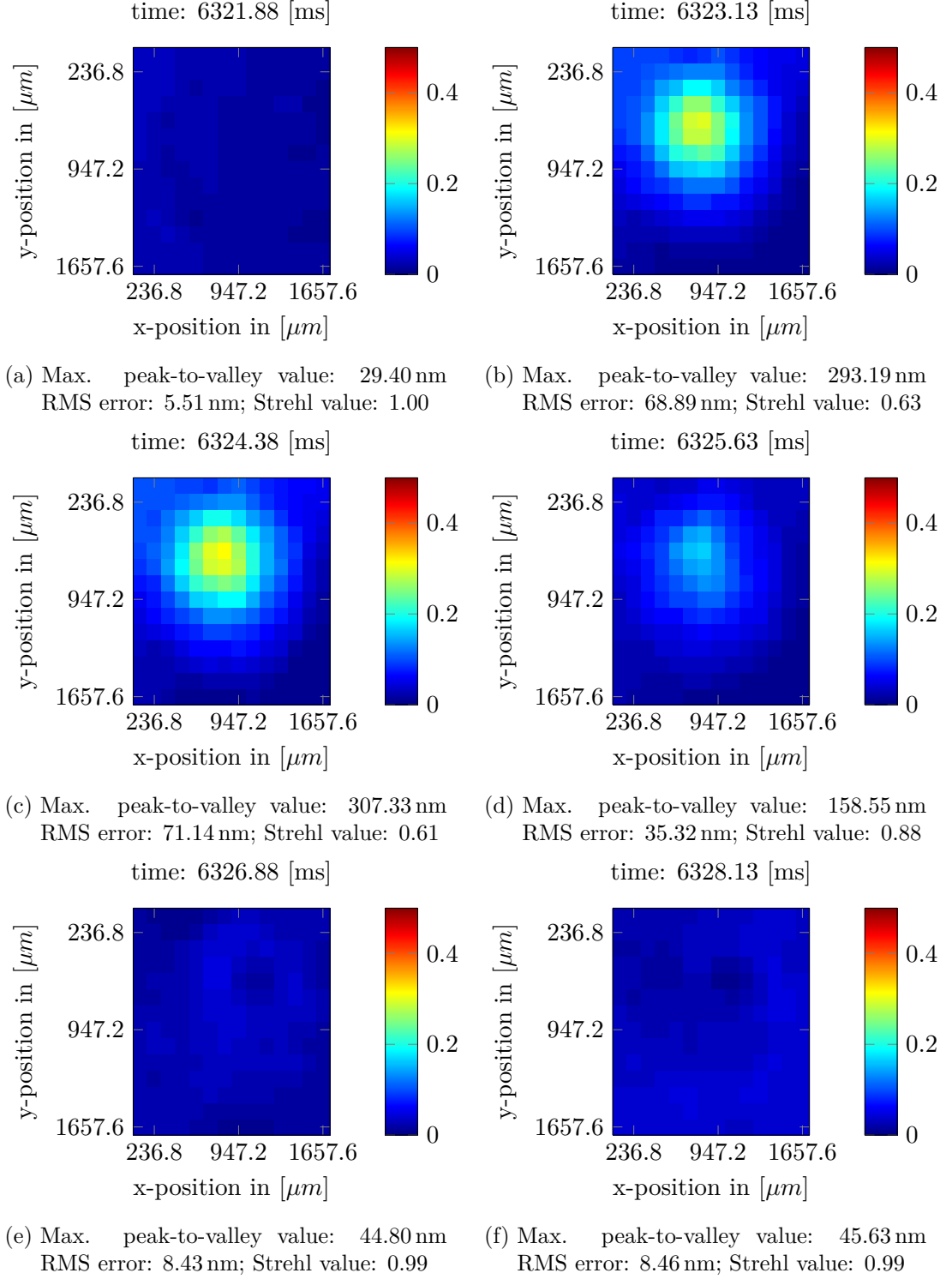


Figure 5.23: Reconstructed wavefront for rejection of step disturbance; same data base as Fig. 5.22

As a side-effect, the wavefront series in Fig. 5.23 demonstrates the correctness of the control matrix since only the slopes of the SHWFS have been used for the modal wavefront reconstruction. As a Gedankenexperiment, it may be assumed that the control matrix is not valid, thus being erroneous. Then, the error value may tend to

zero whereas e.g. the peak-to-valley value of the reconstructed wavefront would not necessarily be as low as it should be. The same is true for the RMS error and Strehl ratio as these quantities are also calculated with the real wavefront and not the error values.

As an intermediate result when comparing the presented experimental with the former simulational results, the outcome is that only a small negligible deviation between the simulation and experiment is recognizable. This deviation may be explained with the limited measurement accuracy of the SHWFS. On the other hand, it is a result of the non-modeled, mostly non-linear effects, e.g. the creep behavior and hysteresis of the piezo actuators.

More sophisticated models for the DM may be identify as well so as to e.g. apply hysteresis compensators [76, 106]. But in underlying context, the expected improvement is very small. The main challenge of guaranteeing stability for the closed-loop is well-treated by using the non-smooth μ -synthesis.

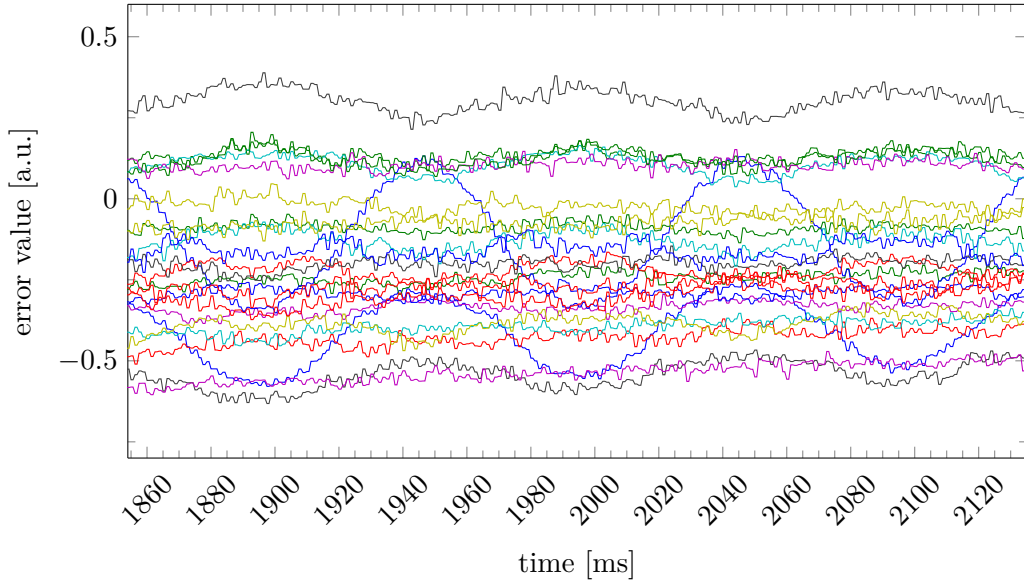


Figure 5.24: Sinusoidal disturbance with 10 Hz uncontrolled (open-loop mode); $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz

Until now only step disturbances have been depicted to judge the control performance of the closed-loop system. For the case of the DM40, additional results when applying disturbances of sinusoidal kind with various frequencies will be presented. Fig. 5.24 shows the result if the disturbance is of sinusoidal type with a frequency of 10 Hz (without enabling the controller, i.e. open-loop mode).

In contrast to step disturbances, sinusoidal disturbances will only partially be rejected at higher frequencies since this kind of signal is not stationary. The amount of damping can be anticipated from the bode plot of the closed-loop (see Fig. 5.16).

During recording of Fig. 5.25, the controller has been activated and the same disturbance as in Fig. 5.24 has been applied. The applied controller is identical to the one used for the simulations and experiments with the rectangular disturbances. The

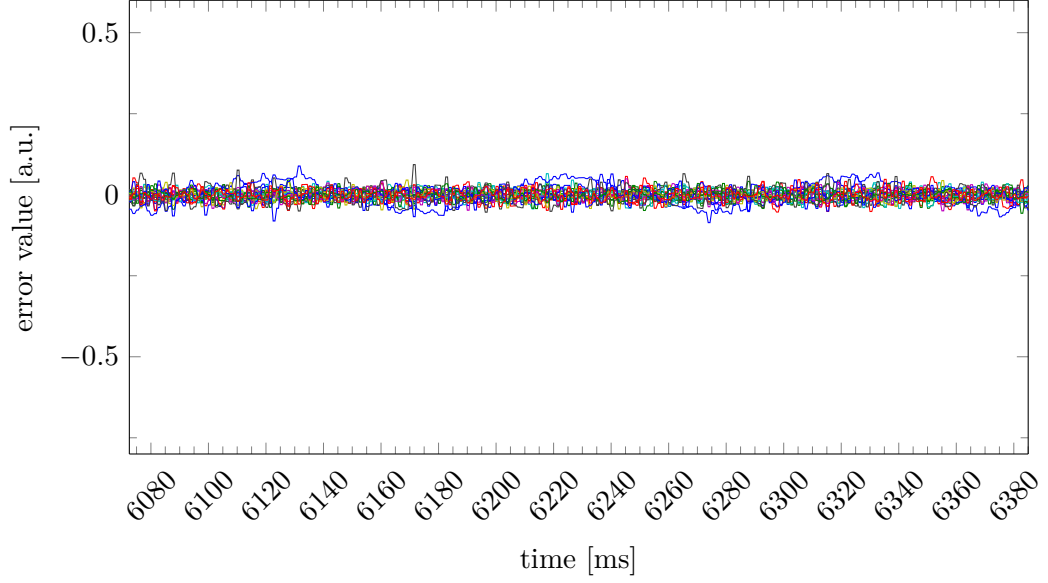


Figure 5.25: Same disturbance as in Fig. 5.24; the controller is switched on (closed-loop mode); $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz

frequency of $f = 10$ Hz is equal to the angular frequency of $\omega \approx 62.83$ rad/s. Based on Fig. 5.16, the damping is expected to be ≈ -17 dB. In the given recorded data of the experiment, the peak-to-valley of the sinus is reduced to ≈ 0.13 .

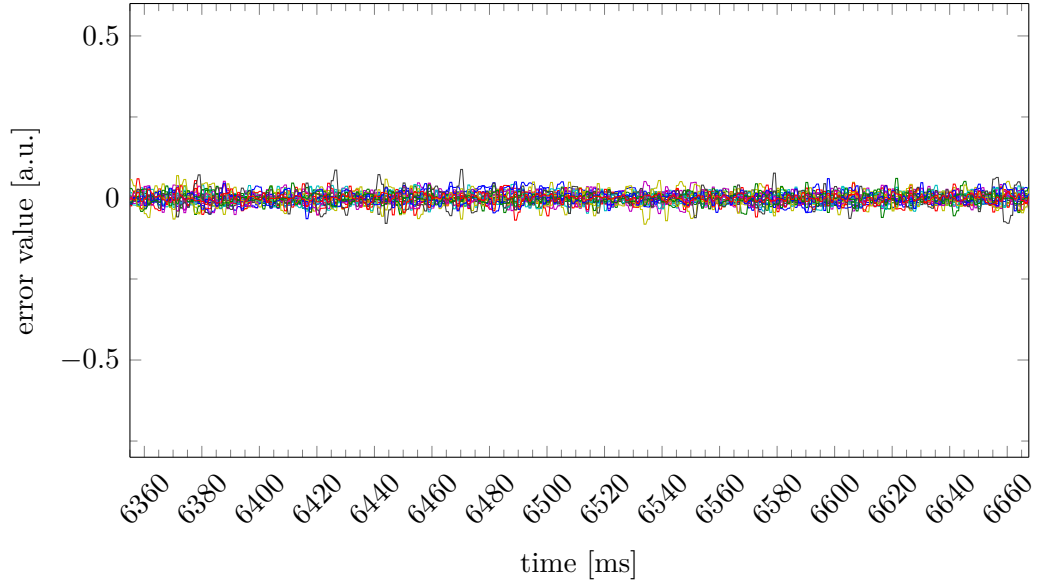


Figure 5.26: Sinus disturbance with 5 Hz with controller activated; $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz

Fig. 5.26 depicts the results when a 5 Hz sinus is applied to an actuator of the disturbance mirror while the controller is enabled. It is obvious that the sinus is rejected such that the error value is free of a sinusoidal signal to large extent. For clarity, the reconstructed wavefront series is not shown.

The next visualized experiment is performed with a frequency of 20 Hz for the sinus

and is given in Fig. 5.27. Due to the increase of frequency of the disturbance, the result is a smaller rejection of the amplitude in the error value compared with the case having a frequency of only 10 Hz. The peak-to-valley of the sinus visible in Fig. 5.27 is ≈ 0.25 .

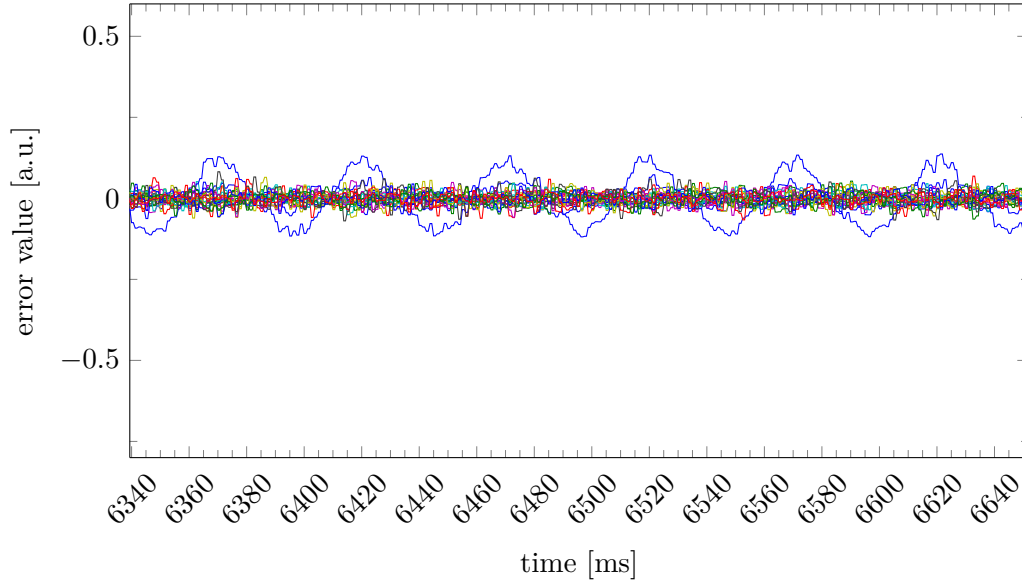


Figure 5.27: Sinus disturbance with 20 Hz with enabled controller; $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz

When the sinus is increased even further to a frequency of 40 Hz, the peak-to-valley increases to ≈ 0.48 in the error value.

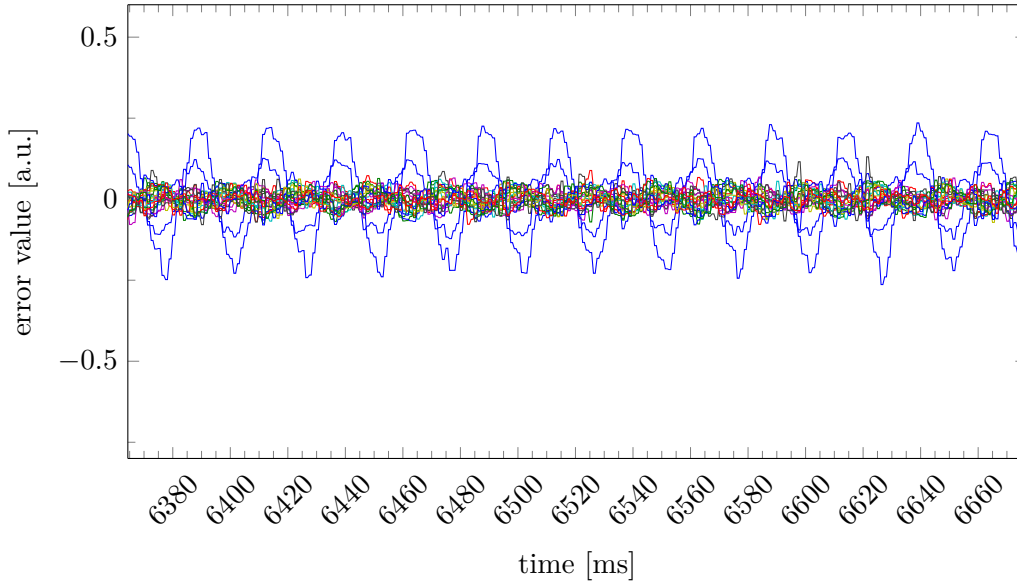


Figure 5.28: Sinus disturbance with 40 Hz with enabled controller; $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz

In Fig. 5.29, the frequency of the sinusoidal signal has increased to the maximum of 80 Hz. The peak-to-valley value increases to ≈ 0.8 . Comparing the result, when

having a 80 Hz sinusoidal disturbance with the bode diagram in Fig. 5.16, the disturbance is amplified by 1 dB. Since the sinusoidal disturbance frequency is reduced to 40 Hz, the peak-to-valley disturbance should be -5.7 dB less than for the case of having a sinusoidal disturbance with 80 Hz. Thus at 40 Hz the amplitude should be ≈ 0.41 whereas Fig. 5.28 is showing a peak-to-valley amplitude of ≈ 0.48 . The remaining peak-to-valley amplitude of the disturbance at 20 Hz must be -12 dB less than for the case of 80 Hz. Thus, the value should be ≈ 0.2 whereas out of Fig. 5.27 the peak-to-valley value may be taken as ≈ 0.25 .

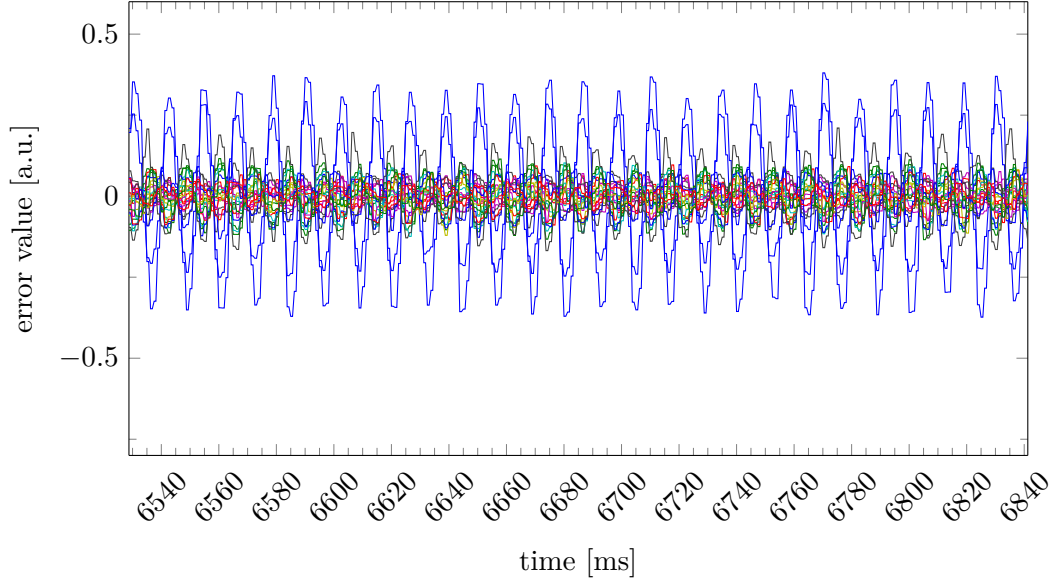


Figure 5.29: Sinus disturbance with 80 Hz with enabled controller; $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz

On the basis of the damping, gathered at various frequencies, it may be stated that a good match between simulation and experiment is obtained, showing only small discrepancies.

\mathcal{H}_∞ optimal PI-controller for DM40

The non-smooth μ -synthesis brings along the advantage that it is capable of synthesizing a structured controller. Hence, the method can also be used to obtain an \mathcal{H}_∞ controller which has a MIMO PI-controller structure.

For this reason, experiments have been conducted with the a synthesis of PI-controllers instead of the state-space \mathcal{H}_∞ controller of order 48. To obtain good results regarding performance and robustness, the weighting functions have been adapted for the synthesis step. Thus, the weighting functions are as follows

$$W_o = 800 \frac{(1/6s + 1)}{(1/1200s + 1)} \quad (5.7)$$

$$W_u = 0.4 \frac{(s/1000 + 1)}{(s/6000 + 1)}. \quad (5.8)$$

The resulting PI-controller does not have the same parameters for each actuator. Instead, for each actuator individual parameters are the output when the non-smooth μ -synthesis is used.

The experimental result when a 10 Hz rectangular disturbance is inserted through the disturbance mirror is shown in Fig. 5.30.

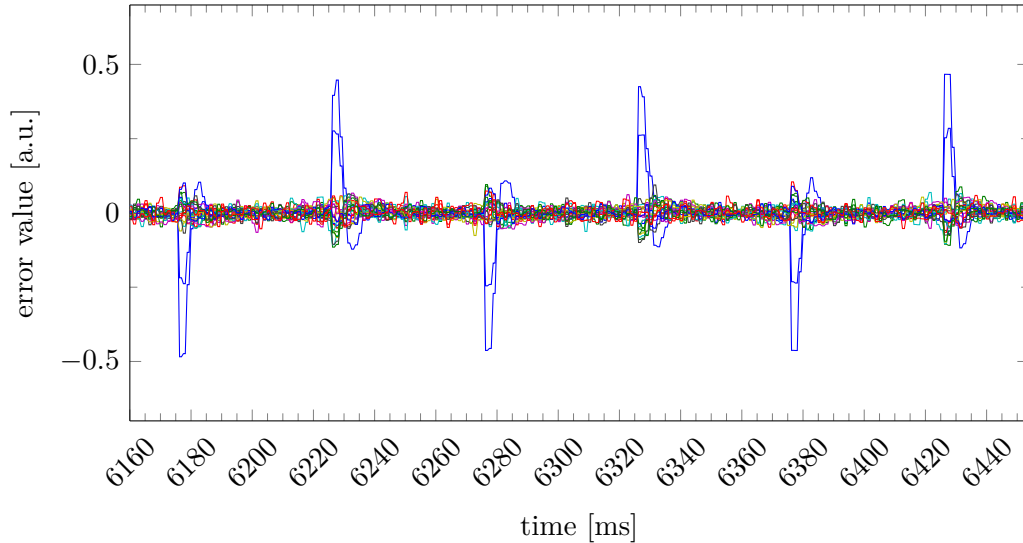


Figure 5.30: Application of a 10 Hz step disturbance with synthesized \mathcal{H}_∞ optimal PI-controller; $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz

Based on Fig. 5.30 it is visible that the result with the synthesized PI-controller is showing some overshoot. This overshoot can be reduced and even eliminated by modifying the weighting matrices accordingly. But as the desired goal here was to achieve a comparable time for disturbance rejection as with the state-space controller of order 48, this overshoot will remain.

The circumstance that the PI-controller of diagonal type has a comparable performance when compared with the state-space controller of order 48 is at first sight somehow surprising. However, what has to be kept in mind is the transformation

of the slopes into actuator values through application of the control matrix. This transformation performs some kind of static decoupling as long as the control matrix is appropriate. Hence, the individual PI-controllers will give a good performance only if the plant is decoupled to some extent. But, since the individual PI-controller for each actuator is independent of the other state, overshooting is much more likely. In terms of stationary accuracy, the PI-controller will have better performance as it incorporates real integral behavior.

6 Conclusion

My momma always said, “Life was like a box of chocolates. You never know what you’re gonna get.”

FORREST GUMP, (1994)

6.1 Overview

In this dissertation, several aspects of adaptive optics (AO) application in high-performance control for wavefront manipulation have been presented. One aspect has been the measuring of the wavefront by using a Shack-Hartmann wavefront sensor (SHWFS). In this regard, novel techniques have been developed in order to significantly improve existing approaches.

To this end, the modified spiral algorithm has been adopted from the literature and successfully implemented on a field programmable gate array (FPGA) for the first time. These implementations are not solely academic as they have also been used in the experiments in a later state. The application of an FPGA turns out very beneficial because it offers far-reaching parallel capabilities. Against this background, the FPGA helps to reduce processing time drastically which then minimizes latency/delay. Due to the implementational effort which occurs when complex algorithms are implemented on an FPGA, the implementation is only feasible for algorithms that are designed in a way that they fully exploit the possibilities of an FPGA, in particular, by parallelism or pipelining.

One important aspect is the evaluation of the image in real-time. This means that the camera image is processed in parallel to the transmission from the camera to the FPGA. Common frame-grabber cards capture the complete image in memory before the data of the image is made public for further processing. In this work, the centroids of the blobs are simultaneously identified during the transmission of the camera data so that in a later step, the centroids can be segmented and assigned to the individual lenses of the lenslet array. This problem could be tackled by using connected component labeling (CCL), applied for the first time in AO. This is a fundamental step when no straight forward assignment between the individual area of the image sensor and the lenses of the lenslet array can be guaranteed.

Afterwards the centroids are assigned to the lenses by a straight-line or the modified spiral ordering approach. These segmentation and ordering approaches also have been successfully implemented and applied on an FPGA which constitutes another novel contribution.

Continuative focus of this dissertation has been put on the control of the deformable

mirror (DM). In this thesis, the DM is not treated as decoupled single input single output (SISO) plants, one per actuator. Instead, the DM is treated as a multiple input multiple output (MIMO) model. Since no model for the expected disturbance is available in the underlying application, \mathcal{H}_∞ methods have been used beneficially in view of attenuating worst-case disturbances while taking additional model uncertainty into account. Finally, non-smooth μ -synthesis has been successfully applied to the MIMO control problem. The non-smooth μ -synthesis yields a structured controller which stabilizes the plant while capturing the model uncertainty at the same time. Since in this approach the controller may be designed supposing a certain controller structure, the result is not necessarily a full-state controller.

Because an existing laboratory virtual instrumentation engineering workbench (LabVIEW) based rapid control prototyping (RCP) system has not been powerful enough for the required calculations, a new RCP concept has been elaborated and realized. This entailed the development of an own peripheral component interconnect express (PCIe) FPGA card in order to be able to assess the wavefront with the novel algorithms for real-time evaluation of the SHWFS. This card also generates the signal for driving the actuators via an interconnected amplifier.

The operating system (OS) used for the RCP system is a hard real-time Linux system, based on real-time application interface (RTAI). A separate driver for using the FPGA card with Linux Control and Measurement Device Interface (comedi) has been developed in order to facilitate the application of MATrix LABoratory (MATLAB) for generating high-performance C-code by means of a Simulink model. Special realizations of Simulink sub-models (s-functions) using the single instruction multiple data (SIMD) instructions of the central processing unit (CPU) have also been developed for the demanded individual acceleration.

Another part of this dissertation is the demonstration and accurate assessment of achievable performance, resulting latencies and the identification of persisting bottlenecks in the setup.

The interaction of the individual processes within high-performance AO control is validated by verification of simulation results with respective laboratory experiments. The achievements are made clear by a significant improvement of controller performance for a DM. The controller design, proposed in this thesis, is made upon a rigorous methodological basis and not in terms of formerly used empirical approaches. Regarding control quality, the outcome is that the elimination of a step disturbance requires approximately five to six frames of the SHWFS. However, comparing the obtained control quality with results achieved by other researchers is difficult since the performance is highly dependent on the DM and SHWFS used in the setup. But it is obvious that increasing the frame-rate of the SHWFS leads to new possibilities, especially for employing different control strategies.

6.2 Future Prospects

- ▶ Developing a new SHWFS camera which offers a higher frame rate and faster transmission of the pixel data. Optionally, the SHWFS may be further accelerated by performing an immediate evaluation of the pixel data in an FPGA which interface the charge-coupled device (CCD)/complimentary metal oxide semiconductor (CMOS) chip directly. The result may be an increased performance and a further reduction of the delay.
- ▶ Incorporating heater and piezoelectric actuators for the compensation of disturbances, i.e. in form of an inner-loop control. The outcome may be an DM which is fully functional with the desired properties. A special feature would be a larger stroke which may especially be beneficial for compensating thermal lenses (mainly defocus) in laser material processing.
- ▶ Cascaded control schemes of the piezoelectric actuators based on integrated strain gauges for linearization; enabling higher performance. Issues here may be the integration of the strain gauges and the signal processing due to expected small changes of resistance in view of the small stroke.
- ▶ Evaluating different control strategies, for example, combining feed-forward control with non-smooth \mathcal{H}_∞ μ -synthesis. A further idea is the integration of anti-windup mechanism in the synthesis process for a-priori actuator limit incorporation.
- ▶ Developing and implementing new methods for real-time transformation of the slopes into an appropriate error signal. The advantage may be an improved handling of ambiguity and nonlinear constraints, such as actuator saturation.

List of Figures

1.1	General adaptive optics (AO) principle	2
2.1	Adaptive optics principle for laser material processing [66]	8
2.2	main characteristic of different DM concepts	9
2.3	Cross-sectional view of the deformable mirror (left side) with buried integrated heaters and temperature sensors. The right side illustrates the rear surface of the mirror with screen-printed actuator structure, integrated heaters and wiring of the temperature sensors. The incoming laser beam is visualized by the red arrows.	10
2.4	Functional principle of the SHWFS in x-direction [67], the red line marks the case when a planar wavefront is sensed with the SHWFS, the black line stands for an exemplarily disturbed wavefront	13
2.5	Single lenslet of an SHWFS	14
2.6	Functional principle of a curvature wavefront sensor (WFS)	16
2.7	Exemplary functional principle of the pyramid wavefront sensor (P-WFS)	17
2.8	Shearing interferometer with small lateral shift ρ	18
2.9	Multiplexing two holograms to detect the given specific aberration	19
2.10	Complete wavefront characterization when two holograms are being multiplexed	19
2.11	Breadboard platform of optical setup for specific disturbance injection and controlling of these disturbances with a deformable mirror	20
2.12	Geometric schematic of optical setup with all telescopes employed	21
3.1	Different reconstruction configurations; the horizontal dots represent the positions of the x-slope and the vertical dots correspond to the y-slope positions. The curls are the points where the phase is estimated.	25
3.2	SHWFS camera image colored in white with reference spots in red (a). The reconstructed wavefront is based on the determined slopes (b).	26
3.3	Illustration of the first six low order Zernike modes	30
3.4	Consequence of a huge positive and negative defocus when having a strict area where the spots must remain	33
3.5	4- and 8-point neighborhood system concerning pixel s	36
3.6	Exemplary blobs sets in a binary image [71]	37
3.7	Label collision due to U shaped object [67]	37
3.8	Example of the centroid ordering algorithm for a 4×4 example [67]	39
3.9	Segmented wavefront image after application of straight line ordering for the case of a huge positive or negative defocus	41
3.10	Spiral algorithm numbering schema [71]	42
3.11	Simulated wavefront image and associated lenses, colored in red [71]	45
3.12	Spot ordering of the wavefront given in Fig. 3.11 using the conventional algorithm: 72 spots failed, the RMS error of the wavefront is in total $2.093 \approx 69.8\%$ [71]	45

3.13	Spot ordering of the wavefront given in Fig. 3.11 using the spiral algorithm: no spots failed, the RMS difference error of the wavefront is in total $0.086 \approx 2.9\%$ [71]	46
3.14	Reconstructed wavefront based on application of conventional (Fig. 3.14a) and spiral algorithm (Fig. 3.14b) for determining spots . .	46
3.15	Difference between global and adaptive thresholding when Gaussian intensity profile is prevalent [65]	47
3.16	Visualization of the nine different possibilities of typically occurring shifts	50
3.17	Centric positioned centroids and the resulting positioning error when new spots (o) have been detected [64]	50
3.18	Labeling of the pixel neighbors for the CCL FPGA implementation [67]	52
3.19	Construction of the pixel stream of the transmitted image when using a 4×4 pixel image sensor	52
3.20	Block diagram of the complete CCL FPGA implementation [67]	53
3.21	Custom build Spartan-6 FPGA board with CameraLink interface . . .	56
3.22	Exemplary test image based on VHDL hardware description language (VHDL) simulations using testbenches	57
3.23	Time evolution with 80 MHz, see Tab. 3.5, (each step rounded up to $50 \mu s$) during image processing	59
3.24	Spiral movement with distance ordering [71]	60
3.25	Simplified overview of the FPGA implementation for the modified spiral algorithm [71]	61
3.26	Time evolution of the complete processing where each step is rounded up to $25 \mu s$ for the modified spiral algorithm [71]	62
3.27	Own-developed PCIe board [71], based on the TE0741 Xilinx Kintex-7 FPGA module from Trenz Electronics [109]	63
3.28	Abstract FPGA pipeline implementation of the adaptive thresholding	64
4.1	Graphical representation of the upper- and lower linear fractional transformation (LFT)	73
4.2	Block diagram of the interconnection of Δ and P in the context of the small gain theorem	74
4.3	Interconnection of $P(s)$ and Δ for calculation of the structured singular value μ	78
4.4	Summary for calculating the output feedback \mathcal{H}_∞ controller	82
4.5	Block diagram with uncertainty, plant and controller	82
4.6	Loop equivalence when the D-scaling transformation is applied for structured controller design	84
4.7	Block diagram of the overall system containing the weighting matrices $W_d(s)$, $W_o(s)$ and $W_u(s)$ and controller $K(s)$	87
4.8	Image of the tip-tilt mirror S-330.2SL from Physik Instruments (PI) .	88
4.9	Bode plot of the measured data, superposed with the identified data for the tip-tilt mirror [68]	89
4.10	Normalized step response of the identified model of the tip-tilt mirror	90
4.11	Step response of the tip-tilt mirror captured by the SHWFS; red curve denotes the step signal, green and blue curve are the reconstructed tip-tilt values, vertical black line marks the one step time-delay	91
4.12	Deformable mirror from Fraunhofer Institute for Applied Optics and Precision Engineering (IOF) with 40 actuators having a diameter of 50 mm	91

4.13	Visualization of the normalized static actuator influence function (AIF) and respective actuator layout/grid of the DM	93
4.14	Measured and identified values for actuator one to eight when deflecting actuator one with an amplitude of 30 V	94
4.15	Measured step response of one piezoelectric actuator via sampling with an SHWFS, red line denotes the stationary value and blue curve the actual measured reconstructed actuator value through the control matrix	95
4.16	Fig. 4.15 partially zoomed such that the behavior around the time of the step is better visible	95
4.17	Normalized step response of the identified model for the DM	96
4.18	Measured hysteresis based on SHWFS measurement and reconstruction through control matrix	96
4.19	Time evolution (each step rounded up to 25 μ s) during SHWFS evaluation by application of the FPGA approach	99
4.20	Time-delay block in the frequency/Laplace domain	102
4.21	Time-delay block from Fig. 4.20 redrawn	102
4.22	Time-delay redrawn using a norm-bounded uncertainty	103
4.23	Phase diagram for $\tau = \{3 \text{ ms}, 4 \text{ ms}\}$ and transfer function $W_u(j\omega) _{\tau=4 \text{ ms}}$	103
5.1	Overview of the LabVIEW based concept used for KD OptiMi - Kompetenz Dreieck Optische Mikrosysteme (KD OptiMi)	107
5.2	Overview over the new control loop concept [69]	111
5.3	Diagram of the PCIe FPGA concept with the separate extension cards	112
5.4	Overview over the complete concept [69] with all involved components	113
5.5	Developed PCIe FPGA card and the additional input/output cards	114
5.6	Installed PCIe FPGA cards in the housing	115
5.7	RTAI principle when RTAI core is active or inactive [69]	115
5.8	Block diagram of the different abstraction layers used in RTAI/LXRT	116
5.9	Invoked modules when an executable for RTAI is built in MATLAB using a Simulink model [69]	117
5.10	Overview over the implemented comedi device driver [69]	118
5.11	Simulated bode diagram with uncertainty (transfer function for tilt disturbance to tilt actuator); red line marks the -3 dB line (intersection is the cut-off frequency)	121
5.12	Simulated step response with uncertainty with Padé-approximation (transfer function for tilt disturbance to tilt actuator)	121
5.13	Simulated step response with uncertainty, noise and no approximation for time-delay (at 100 ms tilt disturbance; at 150 ms tip disturbance)	122
5.14	Step disturbance of 2 Hz with enabled tip-tilt controller; $f_{\text{controller}} = 1600 \text{ Hz}$ and $f_{\text{shwfs}} = 800 \text{ Hz}$	123
5.15	Step disturbance of 2 Hz with enabled tip-tilt controller (zoomed in for comparison with simulation characteristic); $f_{\text{controller}} = 1600 \text{ Hz}$ and $f_{\text{shwfs}} = 800 \text{ Hz}$	124
5.16	Simulated bode plot with uncertainty (transfer function for disturbance one to actuator channel one); red line marks the -3 dB line	126
5.17	Simulated step response with uncertainty (transfer function for disturbance one to actuator channel one)	127
5.18	Simulated step response with uncertainty, noise and no approximation for time-delay (only first two actuators are shown due to clearness)	128

5.19	Visualization of the error value if applying a 2 Hz step disturbance and the controller is not activated (open-loop); sampling rate is $f_{\text{shwfs}} = 800$ Hz	130
5.20	Application of a 10 Hz step disturbance while controller is switched on at $t = 6$ s; $f_{\text{controller}} = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz	130
5.21	Application of a 10 Hz step disturbance with enabled controller; $f_{\text{controller}} = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz	131
5.22	Same experimental data as in Fig. 5.21 but with zoomed x-axis to highlight the control behavior after a step disturbance had occurred, $f_{\text{controller}} = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz	132
5.23	Reconstructed wavefront for rejection of step disturbance; same data base as Fig. 5.22	133
5.24	Sinusoidal disturbance with 10 Hz uncontrolled (open-loop mode); $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz	134
5.25	Same disturbance as in Fig. 5.24; the controller is switched on (closed-loop mode); $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz	135
5.26	Sinus disturbance with 5 Hz with controller activated; $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz	135
5.27	Sinus disturbance with 20 Hz with enabled controller; $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz	136
5.28	Sinus disturbance with 40 Hz with enabled controller; $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz	136
5.29	Sinus disturbance with 80 Hz with enabled controller; $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz	137
5.30	Application of a 10 Hz step disturbance with synthesized \mathcal{H}_∞ optimal PI-controller; $f_s = 1600$ Hz and $f_{\text{shwfs}} = 800$ Hz	138
A.1	Fraunhofer diffraction patten of a rectangular aperture (brightness adapted) ($w_x/w_y = 3$)	154
A.2	Matrix multiplication benchmark for different matrix dimensions; average run-time with 10000 runs in total; stand-alone C++ program, compiled with GCC 4.8.1 and Matlab R2013b running on Linux (identical system configuration) [65]	158
A.3	Latency of RTAI task execution at 10 kHz rate of the LXRT task [65]	159
A.4	Measured times for direct memory access (DMA) transmission of centroids (green line) and image (yellow line)	160
A.5	Measured times for trigger to data valid signal of camera and DMA transmission where exposure time is $100 \mu\text{s}$	161
A.6	Required time consumption for Levenberg non-linear optimization using Eigen3 with RTAI for 24 and 40 actuators	162
A.7	Complete model used for controlling the plant and for synthesizing the application with Simulink Coder	163
A.8	Example of the MATLAB implementation published at Github.com	164
A.9	Scanning Vibrometer PSV-400 from Polytec	165
A.10	Specification of the Scanning Vibrometer PSV-400 [85]	165
A.11	Mounted Imagine Optics HASO TM 3 Fast SHWFS	167
A.12	CCD Truesense KAI-0340DM specification [110]	168
A.13	Imagine Optics HASO TM 3 Fast specification [78]	169

List of Tables

3.1	First six low order normalized Zernike polynomials in polar coordinates	29
3.2	Truth table of the logical AND operator	49
3.3	FPGA logic consumption with $n = 256$, $m = 3$, image width/height 224 pixels (Xilinx Spartan-6 XC6SLX150-3FGG484 FPGA) [67]	58
3.4	FPGA logic consumption with $n = 256$, $m = 3$, image width/height 224 pixels (Xilinx Kintex-7 XC7K160T-2FBG676 FPGA)	58
3.5	Maximum required clock cycles with $n = 256$, $m = 3$ and image width/height 224 pixels [67]	59
3.6	FPGA logic utilization with $n = 256$, $m = 1$, image width/height 224 pixels (Xilinx Kintex-7 XC7K160T-2FBG676 FPGA) when applying the implementing the modified spiral algorithm [71]	62
A.1	Required time for data transfer between PCIe and main memory via DMA [65]	160

Listings

3.1	pseudo code for simple straight centroid ordering [67]	40
-----	--	----

A Appendix

A.1 Fraunhofer Diffraction Pattern

A lenslet array can be regarded as multiple lenses in a particular order. Each lens can be considered as a phase altering component which consists of a different index of refraction than the surrounding.

As an introduction to Fourier Optics for scalar diffraction theory as-well as the Fresnel and Fraunhofer approximation, reference [31] is recommended. Here only a Fraunhofer diffraction pattern for a rectangular aperture [31, chap. 4.4] is considered since some artifacts of the Shack-Hartmann wavefront sensor (SHWFS) may be explained using the Fraunhofer diffraction pattern.

Consider a rectangular aperture given by

$$t_A(\zeta, \eta) = \text{rect}\left(\frac{\zeta}{2w_x}\right) \text{rect}\left(\frac{\eta}{2w_y}\right) \quad (\text{A.1})$$

where w_x and w_y denote half the width of the aperture in the ζ - and η -plane. These planes are parallel to the x-y plane (perpendicular to the z-axis). When the aperture is illuminated by a perpendicular monochromatic plane wave with an amplitude of one, the field distribution over the aperture is equal to the transmittance function t_A . By using the Fraunhofer diffraction pattern [31, chap. 4.3] for rectangular apertures, the following the field strength results in

$$U(x, y) = \frac{e^{jkz} e^{j\frac{k}{2z}(x^2 + y^2)}}{j\lambda z} \mathcal{F}\{U(\zeta, \eta)\} \Big|_{f_x=x/\lambda z; f_y=y/\lambda z}. \quad (\text{A.2})$$

Note that $\mathcal{F}\{U(\zeta, \eta)\} = A \text{sinc}\left(\frac{2w_x x}{\lambda z}\right) \text{sinc}\left(\frac{2w_y y}{\lambda z}\right)$ and the area of the aperture is $A = 4w_x w_y$. Therefore, the intensity for the rectangular aperture is determined by the following relationship

$$I(x, y) = \frac{A^2}{\lambda^2 z^2} \text{sinc}^2\left(\frac{2w_x x}{\lambda z}\right) \text{sinc}^2\left(\frac{2w_y y}{\lambda z}\right). \quad (\text{A.3})$$

The sinc function is defined as follows

$$\text{sinc}(u) = \begin{cases} \frac{\sin(\pi u)}{\pi u} & u \neq 0 \\ 1 & u = 0 \end{cases}. \quad (\text{A.4})$$

Fig. A.1 visualizes the Fraunhofer diffraction pattern of a rectangular aperture when the image sensor is overdriven. Thus, when the image sensor is overdriven (saturated), all intensities above a specific value are limited to the given value. If the image sensor is not saturated then the pattern remains the same, but the intensities of the neighbors are much lower. Therefore, in order to avoid degradation of the accuracy and a strong diffraction pattern, the exposure time is very crucial for good performance of an SHWFS to prohibit image sensor saturation.

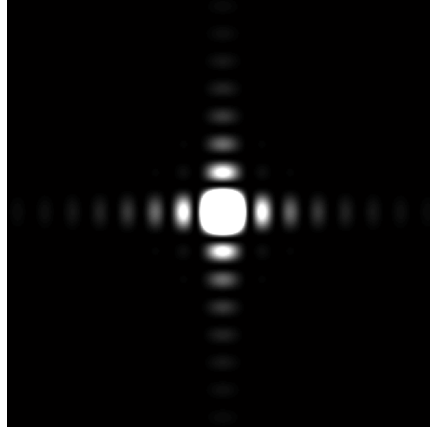


Figure A.1: Fraunhofer diffraction patter of a rectangular aperture (brightness adapted) ($w_x/w_y = 3$)

A.2 Algebraic Riccati Equation and Inequality

Basis for solving standard \mathcal{H}_∞ problems is the Algebraic Riccati Equation (ARE) or the Algebraic Riccati Inequality (ARI). AREs and ARIs are very useful for control system synthesis and therefore play an important role in \mathcal{H}_2 and \mathcal{H}_∞ optimal control [125, 100, 102, 25].

Let A , Q and R denote real $n \times n$ matrices where Q , R are symmetric matrices and R furthermore positive semi-definite. The following matrix equation

$$A^T X + X A + X R X + Q = 0 \quad (\text{A.5})$$

is called an ARE. Adherent with the Riccati equation, a $2n \times 2n$ matrix is associated by the following Hamiltonian matrix

$$H := \begin{bmatrix} A & R \\ -Q & -A^T \end{bmatrix}. \quad (\text{A.6})$$

H is used for obtaining solutions of the ARE. The spectrum of H , $\sigma(H)$, is symmetric regarding the imaginary and real axis [125].

When speaking of an ARI, the following inequality

$$A^T X + X A + X R X + Q < 0 \quad (\text{A.7})$$

is regarded.

Different assumptions on X and Q may be found, depending on the problem formulation and the desired goal. From now on, X is always assumed as real symmetric or complex Hermitian whereas Q is symmetric but indefinite. When solutions of the ARE are desired which fulfill $\lambda(A + RX) \subset \mathbb{C}^-$, the solutions are called stabilizing solutions.

Utilizing the assumption that (A, R) is controllable, the result is the following: The ARE has a solution if and only if the Hamiltonian matrix H , given by A , R , Q , has no eigenvalues on the imaginary axis. If the ARE has a solution, than an unique stabilizing solution X_- and anti-stabilizing solution X_+ exists. All other solutions X of the ARE satisfy $X_- \leq X \leq X_+$.

Theorem 4

Suppose that Q is symmetric, R is positive semi-definite and (A, R) is controllable. Define the Hamiltonian matrix as

$$H := \begin{bmatrix} A & R \\ -Q & -A^T \end{bmatrix},$$

then the following statements are equivalent:

- H has no eigenvalues on the imaginary axis.
- $A^T X + X A + X R X + Q = 0$ has a (unique) stabilizing solution X_- .
- $A^T X + X A + X R X + Q = 0$ has a (unique) anti-stabilizing solution X_+ .
- $A^T X + X A + X R X + Q < 0$ has a symmetric solution X .

If one, hence all, of these conditions are satisfied, then any solution X of the ARE satisfies $X_- \leq X \leq X_+$.

A.3 Bounded Real Lemma

The problem of calculating the \mathcal{H}_∞ norm can be reformulated as a standard linear matrix inequality (LMI) problem by using the Bounded Real Lemma.

Theorem 5 (Bounded Real Lemma)

Consider the system $G(s) = \{A, B, C, D\}$.

The following two conditions are equivalent:

1. A is stable and $\|G\|_\infty < \gamma$
2. There exists $X = X^T > 0$ satisfying

$$\begin{bmatrix} A^T X + X A & X B & C^T \\ B^T X & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} < 0$$

The LMI condition from theorem 5 may be replaced by either

$$\begin{bmatrix} A^T X + X A & X B & C^T \\ B^T X & -\gamma^2 I & D^T \\ C & D & -I \end{bmatrix} < 0$$

or

$$\begin{bmatrix} A^T X + X A + C^T C & X B + C^T D \\ B^T X + D^T C & D^T D - \gamma I \end{bmatrix} < 0$$

or

$$\begin{bmatrix} I & 0 \\ A & B \\ \hline 0 & I \\ C & D \end{bmatrix} \begin{bmatrix} 0 & X & 0 & 0 \\ X & 0 & 0 & 0 \\ \hline 0 & 0 & -\gamma I & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} I & 0 \\ A & B \\ \hline 0 & I \\ C & D \end{bmatrix} < 0$$

or

$$A^T X + X A + C^T C + (X B + C^T D)(\gamma^2 I - D^T D)^{-1}(B^T X + D^T C) < 0 \quad (\text{A.8})$$

Equ. A.8 is the so-called ARI (Appendix A.2).

A.4 Bisection

Bisection is a method for finding roots of a function. The method is based on repeatedly bisecting an interval. The result is then a subinterval where the expected root must lie inside.

The method is very simple and robust, but at the same time relatively slow. Depending on the literature, the method is also called interval halving method or binary search method.

Bisection can be used for computing the critical value γ_{critical} , e.g. for computing the optimal value for the \mathcal{H}_∞ problem or the upper bound for the structured singular value (SSV). In the following, the bisection algorithm which allows to approximate γ is briefly presented:

Given a level of accuracy $\epsilon > 0$:

1. Define an interval $[a_1, b_1]$ such that $a_1 \leq \gamma_{\text{critical}} \leq b_1$.
2. Assume an interval $[a_j, b_j]$ with $a_j \leq \gamma_{\text{critical}} \leq b_j$ has been constructed;
test whether $\gamma_{\text{critical}} < \frac{a_j + b_j}{2}$.
Assuming that this test is feasible leads to an answer of either yes or no.
If the answer is yes, set $[a_{j+1}, b_{j+1}] = [a_j, \frac{a_j + b_j}{2}]$.
If the answer is no, set $[a_{j+1}, b_{j+1}] = [\frac{a_j + b_j}{2}, b_j]$.
3. If $b_{j+1} - a_{j+1} > \epsilon$, then repeat the second step while replacing j with $j + 1$.
If $b_{j+1} - a_{j+1} \leq \epsilon$ then the bisection is stopped with $a_{j+1} \leq \gamma_{\text{critical}} \leq a_{j+1} + \epsilon$.

Since the length of $[a_{j+1}, b_{j+1}]$ is just half of the length of $[a_j, b_j]$, there will always exists an index such that the length of the interval is smaller than ϵ . Therefore the algorithm will always stop. When the algorithm has stopped, γ_{critical} has been calculated up to the absolute accuracy ϵ .

A.5 D/K-Iteration

The structured singular value μ is very useful when the robust performance of a control loop with given plant $P(s)$ and controller $K(s)$ shall be analyzed. However, the structured singular value μ may only be obtain for an already given controller. Up to now, there is no general method which offers a straight-forward way for calculating a controller that minimizes μ [102]. The problem is the following: Finding a controller for the \mathcal{H}_∞ problem is a convex problem, likewise the calculation of μ . However, combining both methods yields a non-convex problem.

Notwithstanding, the so called D/K-iteration, also called scaling/controller iteration, may be applied to combine both tasks and often yields in good results. The convergence of this method, though, cannot be guaranteed in rigorous terms.

One may start with an upper bound on μ

$$\bar{\mu}(\mathcal{F}_\ell(P, K)) \leq \min_{D \in \mathcal{D}} \bar{\sigma}(D \mathcal{F}_\ell(P, K) D^{-1}). \quad (\text{A.9})$$

The basic idea is to find a controller $K(s)$ which minimizes the peak value over all frequencies of the upper bound, that is

$$\min_K \left(\min_{D \in \mathcal{D}} \|D \mathcal{F}_\ell(P, K) D^{-1}\|_\infty \right) \quad (\text{A.10})$$

by alternating between $\|D\mathcal{F}_\ell(P,K)D^{-1}\|_\infty$ with respect to either $K(s)$ or $D(s)$, while the other is held fixed.

For a starting value for $D(s)$, in general, the identity matrix may be used. The algorithm proceeds along the following steps:

1. **K-Step:** Synthesize a \mathcal{H}_∞ controller for some fixed $D(s)$:

$$\min_K \|D\mathcal{F}_\ell(P,K)D^{-1}\|_\infty$$

2. **D-Step:** Find $D(j\omega)$ which minimizes pointwise

$$\bar{\sigma} \left(D(j\omega)\mathcal{F}_\ell(P,K)D(j\omega)^{-1} \right)$$

over all frequencies while fixing $\mathcal{F}_\ell(P,K)$.

3. Fit the magnitude of each element of $D(j\omega)$ to a stable, minimum phase transfer function $D(s)$ and continue with step 1 or stop the procedure.

The iteration may be stopped when a satisfactory performance is achieved, e.g. if $\|D\mathcal{F}_\ell(P,K)D^{-1}\|_\infty < 1$ or when the \mathcal{H}_∞ norm does no longer decrease.

It is important to note that the dimension of the controller $K(s)$ is equal to the number of states plus twice the number of states of $D(s)$. Additionally, due to the bi-convex problem, it is also possible that the minimum is only a local one. For more detailed information regarding the D/K-iteration good starting points are [102, 100].

A.6 Latency/Performance Measurements

In the following, some experimental results regarding latency and performance are given for the real-time Linux system. These experiments have been conducted on the performance computer, consisting of an Intel Core i7 4771 processor (Haswell architecture) mounted on an Asus H87-Pro main-board running Ubuntu 14.04 LTS with a Linux Kernel 3.10.18 patched with RTAI 4.0.

The following software versions have been used for the benchmark: MATrix LABoratory (MATLAB) 2013b, Eigen3 version 3.2.3, matrix multiplication s-function rev. 12/2014 and state-space s-function rev. 12/2014. The peripheral component interconnect express (PCIe) field programmable gate array (FPGA) board rev. 2 with firmware version 1480 has been used. Note that these results have already been presented at the Photonics West 2015 conference [65].

The source-code of the individual Simulink s-functions that have been used is available at <https://github.com/steffenmauch/Simulink-Eigen3>.

A.6.1 Matrix multiplication

In Fig. A.2 the required time for square matrix multiplication of different dimension is visualized. The usual complexity of a matrix multiplication is $O(n^3)$ where n is the dimension of the square matrix.

The so-called schoolbook matrix multiplication itself requires n^3 multiplications and $(n-1)n^2$ additions. From Fig. A.2 it is visible that MATLAB is able to outperform the implemented stand-alone matrix multiplication for large matrix dimensions. This occurs because MATLAB is using the Intel® Math Kernel Library (MKL)¹ which is

¹ Intel® Math Kernel Library, fastest and most used math library for Intel and compatible processors, <https://software.intel.com/en-us/intel-mkl/>

highly optimized to utilize all available capabilities of Intel processors. But for lower matrix dimension, the overhead of MATLAB tampers the results. The streaming SIMD extensions (SSE) and advanced vector extensions (AVX) variant has been coded in C++ using Eigen3. Eigen3 is a C++ template library for linear algebra that was optimized for using the available extensions of the processor. In Fig. A.2, error bars represent minimum and maximum values of the required run-time; for more information see [65].

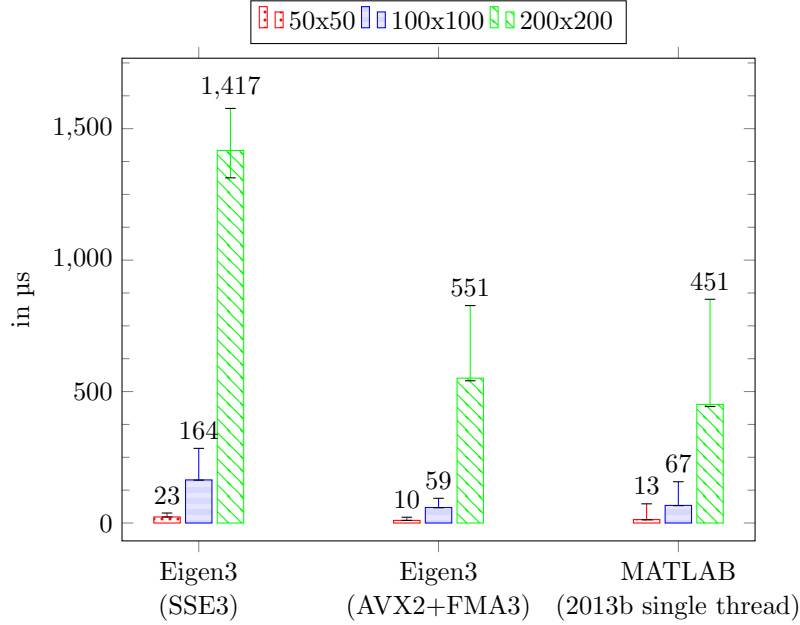


Figure A.2: Matrix multiplication benchmark for different matrix dimensions; average run-time with 10000 runs in total; stand-alone C++ program, compiled with GCC 4.8.1 and Matlab R2013b running on Linux (identical system configuration) [65]

Due to the application of Eigen3 based s-functions for application of the Simulink Coder, even calculations invoking 200×200 matrix multiplications are rendered possible within a loop frequency being larger than 1000 Hz. However, the values from Fig. A.2 are gathered using a stand-alone C++ application.

To demonstrate the validity when using Simulink Coder, a Simulink model has been developed which multiplies a 50×50 matrix. When using the integrated matrix multiplication from Simulink, the time for executing one task step was approximately $140 \mu\text{s}$. Using the Eigen3 based multiplication, the time consumption was only approximately $50 \mu\text{s}$. Of course, these values are distinctively larger than the values in Fig. A.2, but the scheduler overhead and other functionality from RTAI are incorporated in the time consumption as well.

A.6.2 State-Space evaluation

The utilization of the state-space representation for controller or plants has some computational benefits coming along with better numerical stability in comparison when utilizing transfer functions. Furthermore, state-space models may easily be discretized, e.g. by using a rectangle approximation. The only required assumption is that the chosen sampling rate is sufficiently fast.

The state-space equation can be written as follows

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), & x(t_0) &= x_0 \\ y(t) &= Cx(t) + Du(t) \end{aligned} \quad (\text{A.11})$$

where $x(t) \in \mathbb{R}^n$ are the states, $y(t) \in \mathbb{R}^p$ the outputs and $u(t) \in \mathbb{R}^q$ the inputs. Eigen3 is used for reducing the time consumption of the calculation, similar to the matrix multiplication case. The result is in an improvement, but with less pronounced difference in time due to the lower complexity of the required calculations.

The number of required multiplications is $O(n^2 + qn + np + pq)$ while requiring $O(n(n-1) + n(q-1) + p(n-1) + p(q-1))$ additions, in general. Equ. (A.11) can be divided into multiple matrix multiplications dependent on the dimension of $x(t)$ and $u(t)$. For example, for $n = 48$ and $q = 24$, $Bu(t)$ is a matrix multiplication of dimensions $\mathbb{R}^{n \times q}$ by $\mathbb{R}^{q \times 1}$.

A.6.3 RTAI latency

The real-time application interface (RTAI) as-well as the Linux kernel contributes some latency to the overall latency of the SHWFS evaluation. The main difference is that the latency of RTAI and the Linux kernel are load dependent, thus they are not deterministic. Interrupts having a higher priority than the real-time program may influence the latency, i.e. non maskable interrupts of the power management. For having an estimate of the latency, the mean latency over one execution combined with the maximum and minimum (colored blue), as-well as the absolutely worst case (red colored) is given in Fig. A.3. The latency has been determined while the direct memory access (DMA) of the PCIe FPGA card was running (transferring the image as well as the centroids). The processor was forced to 100 % load at all available cores with the program ‘stress’.

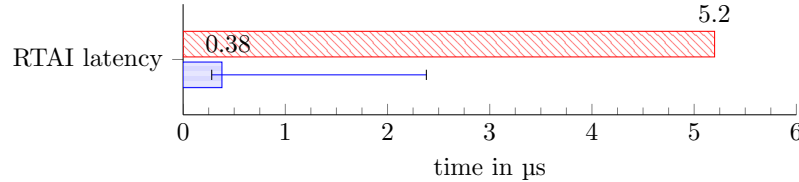


Figure A.3: Latency of RTAI task execution at 10 kHz rate of the LXRT task [65]

During the performance measurement, context switches and X11 server activity have been realized to cover the worst case. In total, the test was carried out over two hours. Recall that latency can be critical even when being low, due to possibility of influencing the integration step. The error directly propagates to the SHWFS because the real-time program triggers the start of a new frame of the SHWFS.

The variance of measured values of the real-time system are negligible because the task execution rate is several times higher than the time-varying fraction of the latency.

A.6.4 DMA transfer from the FPGA to the Computer

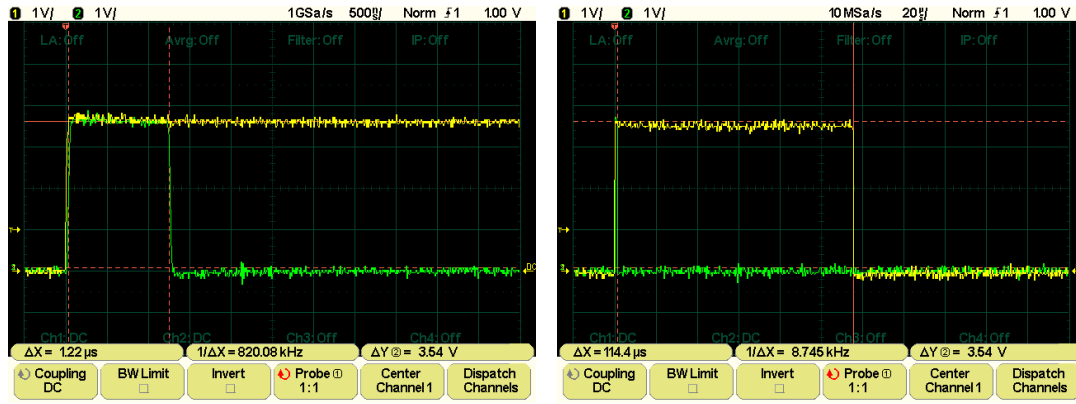
The maximum payload size of the transaction layer packet (TLP) that the chipset of the ASUS main-board is capable allows only a TLP size of 128 bytes. Since each TLP requires an individual header, writing to a 32-bit data address in the main memory of the computer outpace only 116 bytes for the user-data. Thus the actual bandwidth is only $\frac{116}{128} \approx 90\%$ of the burst-rate without the PCIe protocol overhead while neglecting

other possible disturbances.

Tab. A.1 visualizes the required time for transferring the data for the centroids and the whole image. These values are theoretical values but they coincidence with measurements of the corresponding signals within 5-10 % deviation (Fig. A.4).

Table A.1: Required time for data transfer between PCIe and main memory via DMA [65]

data transmission	# of data	req. time PCIe 1.0 x4	req. time PCIe 2.0 x4
centroids	1024 byte	1.08 μ s	0.54 μ s
image	100352 byte	106 μ s	53.1 μ s



(a) PCIe 1.0 x4 - centroids DMA transmission (green line)

(b) PCIe 1.0 x4 - image DMA transmission (yellow line)

Figure A.4: Measured times for DMA transmission of centroids (green line) and image (yellow line)

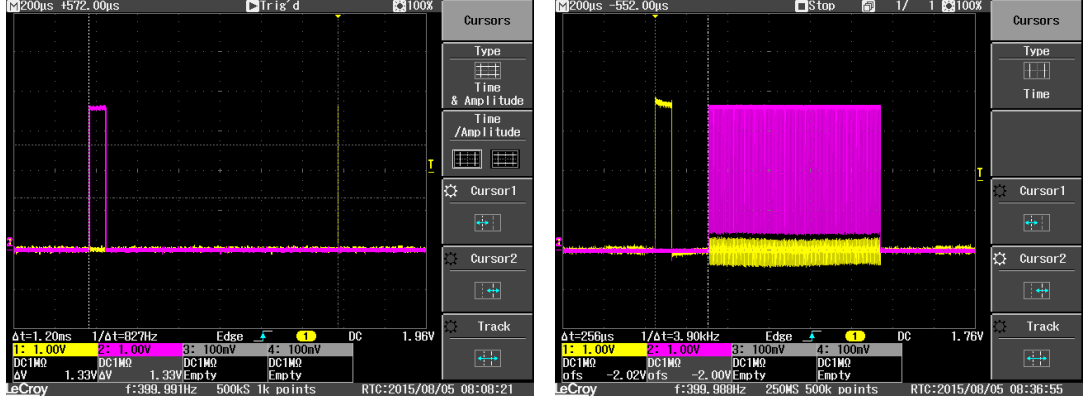
A.6.5 Time Requirement for Trigger to DMA transfer

The time which the evaluation of the SHWFS requires is given in Fig. A.5. The exposure time is 65 μ s. It is visible that the camera requires more than the exposure time to start the transmission of valid data. This behavior occurs as the processor of the camera performs some internal calculation at each trigger signal and the use of blanking rows for image adjustment from the charge-coupled device (CCD) sensor.

Nevertheless the complete required time is less than $\frac{1}{827}$ s while having an exposure time of 65 μ s. Therefore, the delay is smaller than the inverse of the maximum frame rate of the camera in the so-called normal mode.

A.6.6 Slopes to Zernike Conversion

The procedure for conversion of the wavefront slopes into Zernike coefficients has been presented in Sec. 3.1. For different tests, an own s-function that calculates the Zernike based on the slopes has been developed. This s-function is also based on Eigen3 for supporting the vector extensions. It may calculate Zernike coefficients till order 28. The starting point of the implementation is Equ. (3.15) and the corresponding singular



(a) Yellow line DMA transmission, magenta line SHWFS trigger; cursor marks the required time of $\frac{1}{827}$ s in standard mode (b) Yellow line SHWFS trigger, magenta line data valid signal of camera

Figure A.5: Measured times for trigger to data valid signal of camera and DMA transmission where exposure time is $100\mu\text{s}$

value decomposition for solving

$$\Theta = E a. \quad (\text{A.12})$$

Given the singular value decomposition (SVD) with $E = UDV^T$, the Zernike coefficients can be calculated as

$$a = VD^{-1}U^T\Theta. \quad (\text{A.13})$$

It is not necessary, in principle, to compute $VD^{-1}U^T$ for each time instance because only Θ varies. Thus, it is only one matrix multiplication where the size is determined by the number of Zernike coefficients and slopes.

Therefore, the time consumption is $\approx 26\mu\text{s}$ when using Eigen3 with advanced vector extensions 2 (AVX2) and fused multiply-add instruction 3 (FMA3) having 196 slopes for x/y and fitting the first 28 Zernike coefficients.

A.6.7 Non-Linear Optimization for Reconstruction of Actuator Values

In Sec. 4.2.3, the advantage of optimizing a mixture of the \mathcal{L}_1 and \mathcal{L}_2 norm has been discussed for the reconstruction of the control signal for the actuators.

However, it is an optimization process which leads to, in general, non deterministic run-time combined with huge computational effort. So as to evaluate whether the required time for such an optimization is acceptable, an implementation using the single instruction multiple data (SIMD) instruction by application of Eigen3 has been realized. The underlying optimization algorithm is the Levenberg algorithm.

As simulation study, 100000 runs have been performed and the average as well as the maximum and minimum time has been collected. These values are visualized in Fig. A.6.

The studied case represents a worst-case scenario because random values have been used as input of SHWFS slopes data whereas the control matrix is a real captured matrix of the used deformable mirror (DM). An average of approximately 40–50 iterations are required to reach the stop criterion (tolerance of $1e-6$). For the case with 24 actuators as well as 40 actuators, the required time of the implementation is

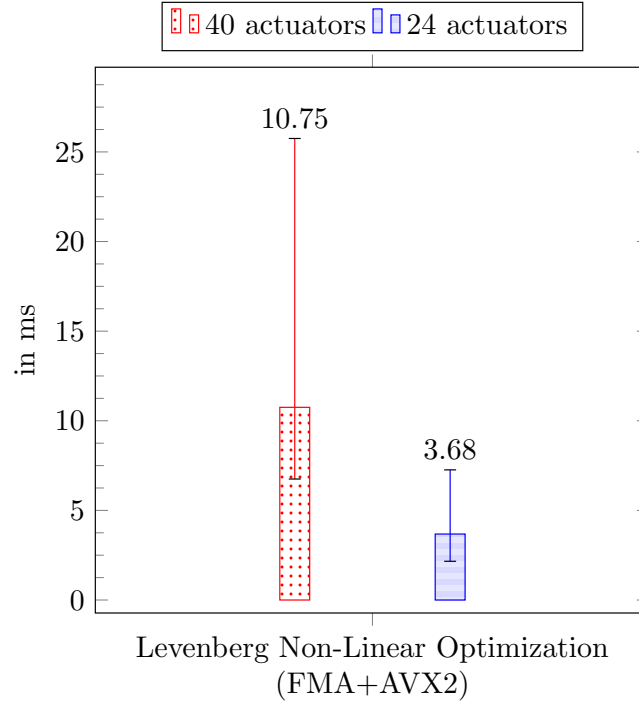


Figure A.6: Required time consumption for Levenberg non-linear optimization using Eigen3 with RTAI for 24 and 40 actuators

too high to be applied in the case of high-performance/speed control.

A.7 Matlab/Simulink Scripts/Models

A.7.1 Control Model Implementation

The Simulink model given in Fig. A.7 was used to synthesize the executable by application of MATLAB/Simulink coder. Its application is executed on the RTAI computer. The input/output blocks are using the Linux Control and Measurement Device Interface (comedi) subsystem for accessing the developed FPGA card whereas the multiplication and controller (state-space block) use the Eigen3² library for fast calculation.

The 'COMEDI_SHWFS_READ' block has as constant input a reference matrix consisting of the centroid position for calculating the slopes. The 'transferMatrix', which is used for the Eigen3 matrix multiplication, is the computed control matrix. Therefore, the actuator influence function (AIF) is captured by a separate Python script which controls the actuator individually and stores the captured centroids into a '.mat' file. Then, based on the AIF, the control matrix is calculated. The wavefront as well as the output signals are stored in a RTAI mailbox ('RTAI_LOG'). An extra Python script (non real-time program) access the mailbox and stores the data in the main memory of the computer. When the predefined main memory is full, the data is written sequentially in a '.mat' file which then is saved on the disk.

² Eigen3 is a C++ template library for calculation of linear algebra problems, which was optimized for using the available extensions of modern processors.

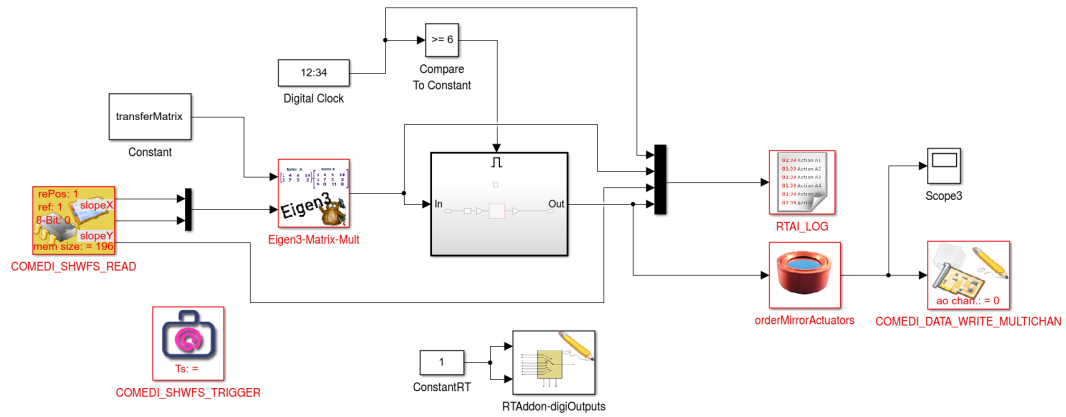


Figure A.7: Complete model used for controlling the plant and for synthesizing the application with Simulink Coder

A.7.2 Wavefront Reconstruction based on SHWFS Slopes

For evaluation purposes of the measured or calculated/simulated centroids or the slopes of an actual SHWFS, a simple zonal reconstruction has been implemented in MATLAB. The method is explained in detail in Sec. 3.1. The implementation as well as a demo of how to use it has been published at <https://github.com/steffenmauch/A0-scripts/tree/master/slopes-visualization>. Fig. A.8 is an example of how the result of the script looks like where the slopes (lower part of the figure) are used to reconstruct the 3D-plot (upper part of the figure).

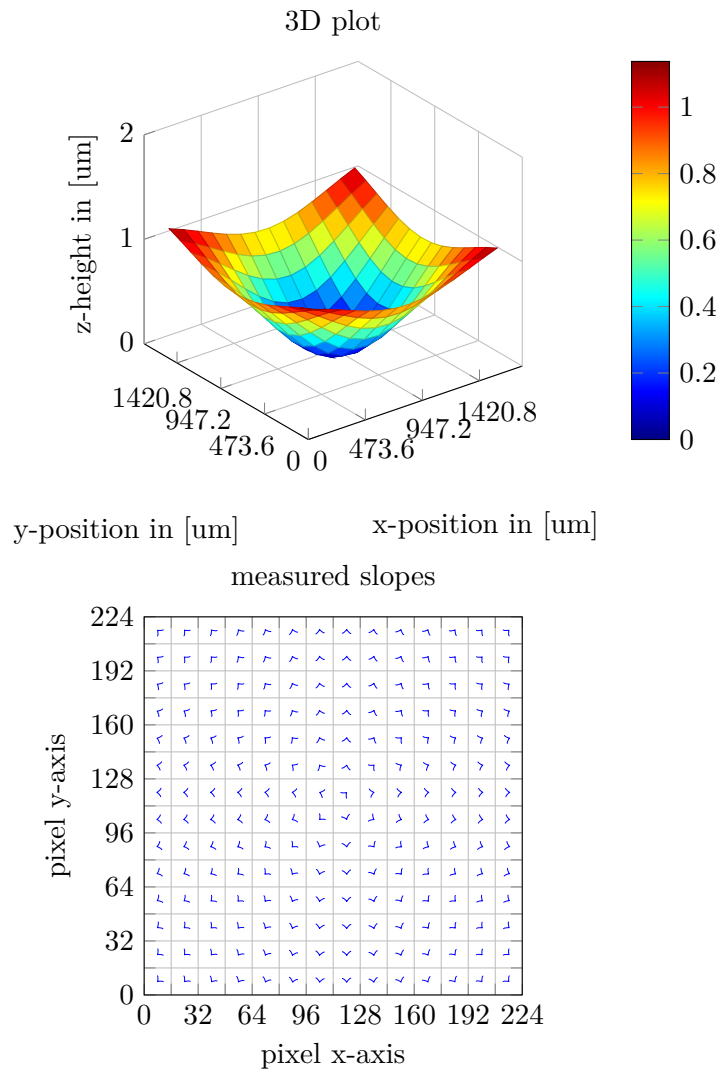


Figure A.8: Example of the MATLAB implementation published at Github.com

A.8 Scanning Vibrometer

The Scanning Vibrometer, also referred as Scanning Laser Vibrometer or Scanning Laser Doppler Vibrometer, is an instrument for fast non-contact measurement and imaging of vibration. The working principle is based on the Doppler effect, which occurs when light is reflected back from a vibrating surface. The velocity and displacement could both be determined by specific analysis of the received optical signals.

The laser scans point-by-point over a surface or generally a test object. A large spatial resolution may thus be covered. The applied vibrometer can cover a range from almost zero to 24 MHz and velocities from $0.02 \frac{\mu\text{m}}{\text{s}}$ to $20 \frac{\text{m}}{\text{s}}$.



Figure A.9: Scanning Vibrometer PSV-400 from Polytec

For the underlying measurements, the Polytec PSV400 Scanning Vibrometer (Fig. A.9) has been applied. The specifications of the device are listed in Fig. A.10.

PSV-400 Decoder/Performance Specifications						
Model	Decoder	# of ranges	Ranges mm s ⁻¹ /V	Full scale (p) m/s	Decoder frequency range	Resolution ¹⁾ $\mu\text{m s}^{-1} / \sqrt{\text{Hz}}$
PSV-400-B	VD-04	3	10 ... 1000	0.1 ... 10	0.5 Hz ... 250 kHz	0.1 ... 5
PSV-400-H4	VD-08	8	0.2 ... 50	0.002 ... 0.5	DC ... 25 kHz	< 0.01 ... 0.2
	VD-09	8	5 ... 1000	0.05 ... 10	DC ... 2.5 MHz	0.01 ... 4
PSV-400-H4-S	VD-03-S	3	20 ... 2000	0.2 ... 20	0.5 Hz ... 1.5 MHz	0.1 ... 5
	VD-07-S	6	2 ... 100	0.02 ... 1	DC ... 350 kHz	< 0.05 ... 0.2
PSV-400-M2	VD-07	6	1 ... 50	0.01 ... 0.5	DC ... 350 kHz	< 0.02 ... 0.2
PSV-400-M4	VD-09	8	5 ... 1000	0.05 ... 10	DC ... 2.5 MHz	0.01 ... 4
PSV-400-M4-S	VD-09-S	14	10 ... 2000	0.1 ... 20	DC ... 2.5 MHz	0.04 ... 8
PSV-400-M2-20 additionally	VD-05 ²⁾	2	100/500	nom. 0.5/2.5	0.5 Hz ... 10 MHz	< 3
	DD-300 ³⁾	1	50 nm/V	75 nm ³⁾	30 kHz ... 24 MHz	< 0.02 pm / $\sqrt{\text{Hz}}$ ⁴⁾
Optional ⁵⁾	DD-900	16	0.05 ... 5000 $\mu\text{m}/\text{V}$	1 μm ... 100 mm ⁶⁾	DC ... 2.5 MHz	< 0.015 ... 1500 nm ⁷⁾

Figure A.10: Specification of the Scanning Vibrometer PSV-400 [85]

Scanning vibrometer measure the velocity of specified points for distinct frequencies of the mirror. In most cases, sinusoidal sweeps are used to measure a broad frequency

band using more than one pass to be able to average the result. With the relation $v = \dot{s}$, where s is the deflection, it is possible to obtain the bode plot of the deflection (shown in Fig. 4.9) for the transfer function $G(s) = \frac{Y(s)}{U(s)}$, where $Y(s)$ is the deflection and $U(s)$ the input voltage of the amplifier.

Measuring the absolute displacement directly is also manageable, but due to accumulating measurement errors, the result will not be reliable as with the velocity measurement.

A.8.1 Identification

Two distinct ways of measuring the deflection of a DM are possible. On the one hand, the deflection may be measured directly when e.g. a step response is exerted on a specific actuator. The problem is that the deflection recorder tends to drift away during the measurement. Since the deflection is in the range of a few nm up to a maximum of $10\text{ }\mu\text{m}$ the drift is not negligible. On the other hand, a step with up to 300 V is not bearable for the piezoelectric actuators of the DM because much mechanical stress occurs.

When the actuators are supplied with a sinus sweep, the velocity may be recorded with high precision. Due to the relationship in the frequency domain of $v = \dot{s}$, where v is the velocity and s the deflection, s can be reconstructed out of measuring v by multiplying with $\omega = 2\pi f$. The Polytec software can directly convert velocity based measurement into deflection and visualize the data in a bode diagram.

First an appropriate grid is defined which captures at least the maxima of the individual actuators on the given surface. Afterwards a sinusoidal sweep is applied to all actuators of the deformable mirror measuring the deflection/velocity of each point of the grid. The resolution of the fast-fourier transformation (FFT) may be changed in the settings of the PSV software. For each actuator, a single ‘svd’ file is generated by the PSV software.

After the capturing step, for each ‘svd’ file, the bode plot is selected in the PSV software and the data is exported as American Standard Code for Information Interchange (ASCII) file to be further processed with MATLAB. If there are more measurement points than actuators have been saved previously, the index number of the actuators have to be identified such that the exported ASCII files may be processed correctly based on the index number.

For each actuator, the deflection at the other actuators are captured as well such that the model identification can be started. Of course, the measurements are only valid when the system is linear. If this is not the case, the frequency data will not be accurate or even totally wrong. At least, the system must be regarded as ‘linear enough’ such that the measurement makes sense for the identification.

A.9 Imagine Optics HASOTM3 Fast

The SHWFS HASOTM3 Fast from Image Optics is based on an IMPERX Bobcat ICL-B0620M camera. The CCD sensor mounted inside the camera is a Truesense KAI-0340DM offering a resolution of $648/640 \times 488/480$ with a maximum speed of 210/260 fps. The camera captures monochrome images with a pixel resolution of either 8/10/12/14 bit (see specification in Fig. A.12).

The HASOTM3 Fast has a 14×14 lens/aperture array when using the supposed 224×224 pixels. Using the 224×224 pixels of resolution is the intended use-case of Imagine Optics. Thus, the frame-rate is 905 frames per second. For the applied HASOTM3 Fast, each lens of the lenslet array has a focal length 4.796 mm, but other focal lenses are available from Imagine Optics as well. The specification of the SHWFS stated by Imagine Optics are listed in Fig. A.13.

Using the HASO software from Imagine Optics, basically, the settings of the camera cannot be altered except for the exposure time. The HASO software offers an automatic control of the exposure time when the intensity of the laser beam will change over time. Nevertheless, using the Bobcat configurator offered by IMPERX, the camera is almost fully customizable requiring only a Windows computer and access to the serial port which is transmitted over the CameraLink interface.

Based on the selected pixel resolution, two different modes are available: single-tap and dual tap. The difference is that in single-tap mode only one pixel per clock cycle is transmitted while in dual-tap mode, two pixel per clock cycle are transmitted. For the interface, CameraLink base standard is used. Therefore, when using the 14 bit mode, only single-tap is possible because the number of available bits per pixel clock is limited to 28 bits in total and the line-, frame- and data-valid also has to be transmitted.

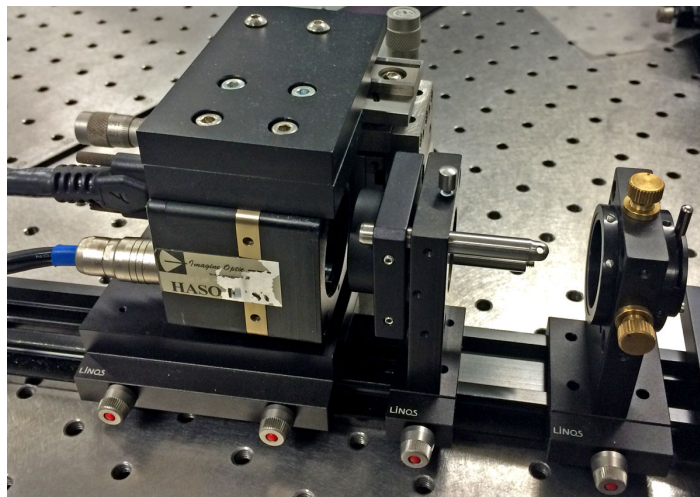


Figure A.11: Mounted Imagine Optics HASOTM3 Fast SHWFS

When reducing the resolution of the camera image, as employed by Imagine Optics, the frame-rate may be increased. For example, a resolution of 224×224 pixels with up to 905 frames per second is possible, having either 12/10 or 8 bit of intensity information. Additionally, over-clocking of the CCD is possible as well but increases the pixel noise which leads to less resolution depending on the application in low-light settings.

The base pixel clock of the CameraLink interface is 40 MHz, which is an impor-

Parameter	Value
Architecture	Interline CCD; Progressive Scan
Total Number of Pixels	696 (H) x 492 (V)
Number of Effective Pixels	648 (H) x 484 (V)
Number of Active Pixels	640 (H) x 480 (V)
Pixel Size	7.4 μm (H) x 7.4 μm (V)
Active Image Size	4.736mm (H) x 3.552mm (V) 5.920mm (diagonal) 1/3" optical format
Aspect Ratio	4:3
Number of Outputs	1 or 2
Charge Capacity	40 MHz – 20,000 electrons 20 MHz – 40,000 electrons
Output Sensitivity	30 $\mu\text{V}/\text{e}$
Photometric Sensitivity KAI-0340-ABB	3.61 V/lux-sec
Photometric Sensitivity KAI-0340-CBA	1.17(B), 1.54(G), 0.65(R) V/lux-sec
Readout Noise	40 MHz – 16 electrons 20 MHz – 14 electrons
Dynamic Range	40 MHz – 62 dB 20 MHz – 69 dB
Dark Current	Photodiode < 200 eps VCCD < 1000 eps
Maximum Pixel Clock Speed	40MHz
Maximum Frame Rate	KAI-0340-Dual – 210 fps KAI-0340-Single – 110 fps
Package Type	CerDIP
Package Size	0.500" [12.70mm] width 0.625" [15.87mm] length
Package Pins	22
Package Pin Spacing	0.050"

All parameters above are specified at T = 40 °C

Figure A.12: CCD Truesense KAI-0340DM specification [110]

tant characteristic when the camera is connected to the FPGA. When using dual-tap transmission, the FPGA must either operate on two pixels in parallel or the connected component labeling (CCL) or the logic must be able to operate at 80 MHz such that the pixel may be processed sequentially without being stored. The ICL-B0620M offers a lot of different tunable parameters which are well-described in the ‘Bobcat Hardware User Manual’.

One important option, which is used extensively, is the trigger option. The camera may be triggered over the CameraLink interface such that the frame-rate is synchronized with the FPGA and thus, the computer system. Therefore, time-varying delays, which likely happen when systems are not synchronized, are circumvented. Regardless of the mode of operation, one has to ensure that the exposure time is appropriate. Otherwise, when the exposure time is too long, the intensity of the pixel will be clipped, resulting in the fact that the brightness value is limited to the maximum, e.g. 256 having 8 bit intensity information. Hence, valuable information will be lost and the accuracy of the centroid position determination is influenced. On the contrary, when the exposure time is too low, the signal-to-noise ratio (SNR) will be lower than necessary which is also leading to a worse accuracy.

Aperture dimension	1.7 x 1.7 mm ²
Number of microlenses	14 x 14
Refractive microlens technology	standard square
Tilt dynamic range	$> \pm 3^\circ$
Focus dynamic range - minimum local radius of curvature	± 0.025 m
Focus dynamic range - maximum NA	0.1
Repeatability (rms)	$< \lambda/200$
Wavefront measurement accuracy in absolute mode rms ¹	$\sim \lambda/100$
Wavefront measurement accuracy in relative mode rms ²	$\sim \lambda/150$
Tilt measurement sensitivity (rms)	6 μ rad
Focus measurement sensitivity (rms)	5×10^{-3} m ⁻¹
Spatial resolution	~ 110 μ m
Maximum acquisition frequency	905 Hz
Processing frequency	800 Hz
Working wavelength	350-1100 nm
Calibrated wavelength band	On request
Extended calibrated wavelength band	On request
Dimensions / weight	115 x 51 x 60 mm / 400 g
Working temperature	15 – 30°
Power supply	12 V / 6 W
Interface	CamLink
1) Difference between the real wavefront and a reference wavefront obtained in similar conditions (10 λ of shift maximum). 2) Wavefront as seen by the wavefront sensor. Performance kept on the whole spectral range.	

Figure A.13: Imagine Optics HASOTM3 Fast specification [78]

A.10 Simulation and Experimental Data

For reproducing the simulations, required data such as the values for the controller and identified model of the DM are available either on the enclosed CD-ROM or via <https://github.com/steffenmauch/PhD-AdaptiveOptics>.

A.11 List of Own Publications

- N. Leonhard, R. Berlich, S. Minardi, A. Barth, S. Mauch, J. Mocci, M. Goy, M. Appelfelder, E. Beckert, and C. Reinlein, “Real-time adaptive optics testbed to investigate point-ahead angle in pre-compensation of earth-to-geo optical communication,” *Opt. Express*, vol. 24, no. 12, pp. 13 157–13 172, Jun 2016. DOI: <http://dx.doi.org/10.1364/OE.24.013157>
- S. Mauch, “Controlled Impedance for Edge-Coupled Coated Microstrip with Cadsoft Eagle V6,” *Embedded Projects*, vol. 3/2013, pp. 20–21, 2013.
- S. Mauch, “Overcome Fear of Ethernet on an FPGA,” *Circuit Cellar*, vol. CC281, pp. 20–25, 2013.
- S. Mauch, A. Barth, J. Reger, N. Leonhard, and C. Reinlein, “Improved Thresholding and Ordering for Shack-Hartmann wavefront sensors implemented on an FPGA,” *International Workshop on Adaptive Optics for Industry and Medicine (AOIM)*, 2015.
- S. Mauch, A. Barth, J. Reger, C. Reinlein, M. Appelfelder, and E. Beckert, “FPGA-accelerated adaptive optics wavefront control part II,” *Proceedings SPIE*, vol. 9343, pp. 93 430Y–93 430Y–13, 2015. DOI: <http://dx.doi.org/10.1117/12.2079010>
- S. Mauch and J. Reger, “Application of μ -Synthesis H_∞ -Control for Adaptive Optics in Laser Material Processing,” in *2013 IEEE International Conference on Control Applications (CCA)*, 2013, pp. 941–947. DOI: <http://dx.doi.org/10.1109/CCA.2013.6662872>
- S. Mauch and J. Reger, “Real-time spot detection and ordering for a Shack-Hartmann wavefront sensor with a low-cost FPGA,” *IEEE Transactions on Instrumentation and Measurement*, 2014. DOI: <http://dx.doi.org/10.1109/TIM.2014.2310616>
- S. Mauch, J. Reger, and E. Beckert, “Adaptive Optics control for laser material processing,” *Congreso Latinoamericano de Control Automático (CLCA 2012)*, Peru, Lima, 2012. [Online]. Available: https://www.tu-ilmenau.de/fileadmin/public/regelungstechnik/Mitarbeiter/Prof_Johann_Reger/reger_CLCA2012.pdf
- S. Mauch, J. Reger, C. Reinlein, M. Appelfelder, M. Goy, E. Beckert, and T. Tünnermann, “FPGA-accelerated adaptive optics wavefront control,” *Proceedings SPIE*, vol. 8978, pp. 1–12, 2014. DOI: <http://dx.doi.org/10.1117/12.2038910>
- S. Mauch, “Neuartiges Verfahren basierend auf einem FPGA für die Auswertung eines Shack-Hartmann-Wellenfrontsensors,” german Patent 10 2014 017 050 A1, 12 18, 2013. [Online]. Available: <http://www.google.com/patents/DE102014017050A1?cl=de>
- S. Mauch and J. Reger, “Real-Time Implementation of the Spiral Algorithm for Shack-Hartmann Wavefront Sensor Pattern Sorting on an FPGA,” *Measurement*, pp. –, 2016. DOI: <http://dx.doi.org/10.1016/j.measurement.2016.06.004>

B Bibliography

- [1] E. Acosta, S. Bará, M. A. Rama, and S. Ríos, “Determination of phase mode components in terms of local wave-front slopes: an analytical approach,” *Optics Letters*, vol. 20, no. 10, pp. 1083–1085, May 1995. DOI: <http://dx.doi.org/10.1364/OL.20.001083>
- [2] L. Adams, “Cosmic ray effects in microelectronics,” *Microelectronics Journal*, vol. 16, no. 2, pp. 17 – 29, 1985. DOI: [http://dx.doi.org/10.1016/S0026-2692\(85\)80213-5](http://dx.doi.org/10.1016/S0026-2692(85)80213-5)
- [3] O. Albert, L. Sherman, G. Mourou, T. B. Norris, and G. Vdovin, “Smart microscope: an adaptive optics learning system for aberration correction in multiphoton confocal microscopy,” *Optics Letters*, vol. 25, no. 1, pp. 52–54, Jan 2000. DOI: <http://dx.doi.org/10.1364/OL.25.000052>
- [4] G. P. Andersen, L. Dussan, F. Ghebremichael, and K. Chen, “Holographic wave-front sensor,” *Optical Engineering*, vol. 48, no. 8, pp. 085 801–085 801–5, 2009. DOI: <http://dx.doi.org/10.1117/1.3204232>
- [5] G. P. Andersen, F. Ghebremichael, R. Gaddipati, P. Gaddipati, P. Gelsinger-Austin, and K. MacDonald, “Fast computer-free holographic adaptive optics,” *Proceedings SPIE*, vol. 8744, pp. 87 440Y–87 440Y–6, 2013. DOI: <http://dx.doi.org/10.1117/12.2015900>
- [6] L. C. Andrews, *Field Guide to Atmospheric Optics (SPIE Vol. FG02)*. SPIE Publications, 1 2004.
- [7] P. Apkarian and D. Noll, “Nonsmooth H_∞ synthesis,” *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 71–86, Jan 2006. DOI: <http://dx.doi.org/10.1109/TAC.2005.860290>
- [8] P. Apkarian, “Nonsmooth μ -synthesis,” *International Journal of Robust and Nonlinear Control*, vol. 21, no. 13, pp. 1493–1508, 2011. DOI: <http://dx.doi.org/10.1002/rnc.1644>
- [9] P. Apkarian. (2012, Sep.) Examples for nonsmooth μ synthesis with Matlab. [Online]. Available: <http://pierre.apkarian.free.fr/>
- [10] P. Apkarian and D. Noll, “Nonsmooth optimization for multidisk H_∞ synthesis,” *European Journal of Control*, vol. 12, no. 3, 2006. DOI: <http://dx.doi.org/10.3166/ejc.12.229-244>
- [11] P. Apkarian and D. Noll, “The H_∞ control problem solved,” *website*, pp. 1–27, 2012. [Online]. Available: <http://www.math.univ-toulouse.fr/~noll/PAPERS/solved.pdf>
- [12] P. Apkarian and D. Noll, “Optimization-based control design techniques and tools,” in *Encyclopedia of Systems and Control*, J. Baillieul and T. Samad, Eds. Springer London, 2014, pp. 1–12. DOI: http://dx.doi.org/10.1007/978-1-4471-5102-9_144-1
- [13] D. Bailey, C. Johnston, and N. Ma, “Connected components analysis of streamed images,” in *International Conference on Field Programmable Logic and Ap-*

- plications*, 2008, pp. 679–682. DOI: <http://dx.doi.org/10.1109/FPL.2008.4630038>
- [14] A. G. Basden, F. Assémat, T. Butterley, D. Geng, C. D. Saunter, and R. W. Wilson, “Acceleration of adaptive optics simulations using programmable logic,” *Monthly Notices of the Royal Astronomical Society*, vol. 364, no. 4, pp. 1413–1418, 2005. DOI: <http://dx.doi.org/10.1111/j.1365-2966.2005.09670.x>
- [15] R. J. Beck, *Adaptive optics for laser processing*. Heriot-Watt University, 10 2011. [Online]. Available: <http://hdl.handle.net/10399/2462>
- [16] P. Bedggood and A. Metha, “Comparison of sorting algorithms to increase the range of Hartmann-Shack aberrometry,” *Journal of Biomedical Optics*, vol. 15, no. 6, pp. 067 004–067 004–7, 2010. DOI: <http://dx.doi.org/10.1117/1.3516706>
- [17] M. J. Booth, “Adaptive optics in microscopy,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 365, no. 1861, pp. 2829–2843, 2007. DOI: <http://dx.doi.org/10.1098/rsta.2007.0013>
- [18] I. Buske, “Aberrationen in Nd:YAG - Hochleistungslasern und -verstärkern: Ihr Einfluss und ihre Korrektur mit adaptiver Optik,” *Dissertation TU Berlin*, 2005. [Online]. Available: https://opus4.kobv.de/opus4-tuberlin/files/1078/buske_ivo.pdf
- [19] W. J. Chaohong Li, Hao Xian and C. Rao, “Measurement error of Shack-Hartmann Wavefront Sensor,” *Topics in Adaptive Optics*, 2012. DOI: <http://dx.doi.org/10.5772/29430>
- [20] M. Claire, “Introduction to adaptive optics and its history,” *American Astronomical Society, 197th Meeting*, 2001.
- [21] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. McGraw-Hill Science/Engineering/Math, 12 2003.
- [22] G.-M. Dai, “Comparison of wavefront reconstructions with zernike polynomials and fourier transforms.” *Journal of Refractive Surgery*, vol. 22, no. 9, pp. 943–8, 2006.
- [23] J. Doyle and G. Stein, “Multivariable feedback design: Concepts for a classical/-modern synthesis,” *IEEE Transactions on Automatic Control*, vol. 26, no. 1, pp. 4–16, Feb 1981. DOI: <http://dx.doi.org/10.1109/TAC.1981.1102555>
- [24] J. Doyle, “Analysis of feedback systems with structured uncertainties,” *IEEE Proceedings on Control Theory and Applications*, vol. 129, no. 6, pp. 242–250, November 1982. DOI: <http://dx.doi.org/10.1049/ip-d.1982.0053>
- [25] G. E. Dullerud and F. Paganini, *A Course in Robust Control Theory: A Convex Approach (Texts in Applied Mathematics)*. Springer, 2 2005.
- [26] D. L. Fried, “Least-square fitting a wave-front distortion estimate to an array of phase-difference measurements,” *Journal of the Optical Society of America*, vol. 67, no. 3, pp. 370–375, Mar 1977. DOI: <http://dx.doi.org/10.1364/JOSA.67.000370>
- [27] P. Gahinet and P. Apkarian, “Structured H_∞ Synthesis in MATLAB,” *18th IFAC World Congress*, 2011. DOI: <http://dx.doi.org/10.3182/20110828-6-IT-1002.00708>

-
- [28] P. Gahinet and P. Apkarian, “The H_∞ Synthesis in MATLAB,” *18th IFAC World Congress*, 2011. DOI: <http://dx.doi.org/10.3182/20110828-6-IT-1002.00708>
 - [29] A. Gavrielides, “Vector polynomials orthogonal to the gradient of zernike polynomials,” *Optics Letters*, vol. 7, no. 11, pp. 526–528, Nov 1982. DOI: <http://dx.doi.org/10.1364/OL.7.000526>
 - [30] F. Ghebremichael, G. P. Andersen, and K. S. Gurley, “Holography-based wave-front sensing,” *Applied Optics*, vol. 47, no. 4, pp. A62–A69, Feb 2008. DOI: <http://dx.doi.org/10.1364/AO.47.000A62>
 - [31] J. Goodman, *Introduction to Fourier Optics*, 3rd ed. Roberts and Company Publishers, 12 2004.
 - [32] S. J. Goodsell, D. Geng, E. Fedrigo, C. Soenke, R. Donaldson, C. D. Saunter, R. M. Myers, A. G. Basden, and N. A. Dipper, “Fpga developments for the sparta project: Part 2,” *Proceedings SPIE*, vol. 6272, pp. 627 241–627 241–12, 2006. DOI: <http://dx.doi.org/10.1117/12.671646>
 - [33] D. P. Greenwood, “Bandwidth specification for adaptive optics systems*,” *J. Opt. Soc. Am.*, vol. 67, no. 3, pp. 390–393, Mar 1977. DOI: <http://dx.doi.org/10.1364/JOSA.67.000390>
 - [34] D. P. Greenwood and C. A. Primmerman, “Adaptive Optics Research at Lincoln Laboratory,” *The Lincoln Laboratory Journal Vol 5, Num. 1*, 1992. [Online]. Available: https://www.ll.mit.edu/publications/journal/pdf/vol05_no1/5.1.1.opticsresearch.pdf
 - [35] Y. Guo, C. Rao, H. Bao, A. Zhang, X. Zhang, and K. Wei, “Multichannel-hadamard calibration of high-order adaptive optics systems,” *Optics Express*, vol. 22, no. 11, pp. 13 792–13 803, Jun 2014. DOI: <http://dx.doi.org/10.1364/OE.22.013792>
 - [36] N. Gutzeit, J. Müller, C. Reinlein, and S. Gebhardt, “LTCC membranes With integrated heating structures, temperature sensors and strain gauges,” in *35th International Spring Seminar on Electronics Technology (ISSE)*, 2012, pp. 399–405. DOI: <http://dx.doi.org/10.1109/ISSE.2012.6273169>
 - [37] N. Gutzeit, J. Müller, C. Reinlein, and S. Gebhardt, “Manufacturing and Characterization of a Deformable Membrane with Integrated Temperature Sensors and Heating Structures in Low Temperature Co-fired Ceramics,” *International Journal of Applied Ceramic Technology*, vol. 10, no. 3, pp. 435–442, 2013. DOI: <http://dx.doi.org/10.1111/ijac.12037>
 - [38] O. Guyon, “Limits of Adaptive Optics for High-Contrast Imaging,” *The Astrophysical Journal*, vol. 629, no. 1, p. 592, 2005. DOI: <http://dx.doi.org/10.1117/12.789849>
 - [39] J. W. Hardy, *Adaptive Optics for Astronomical Telescopes*. Oxford University Press, Jul. 1998.
 - [40] P. Hickson, “Fundamentals of Atmospheric and Adaptive Optics,” *Homepage University of British Columbia*, 2008. [Online]. Available: www.aeos.ulg.ac.be/upload/ao.pdf
 - [41] R. H. Hudgin, “Wave-front reconstruction for compensated imaging,” *Journal of the Optical Society of America*, vol. 67, no. 3, pp. 375–378, Mar 1977. DOI: <http://dx.doi.org/10.1364/JOSA.67.000375>
 - [42] J. Ion, *Laser Processing of Engineering Materials: Principles, Procedure and Industrial Application*, 1st ed. Butterworth-Heinemann, 5 2005.

- [43] R. Irwan and R. G. Lane, "Analysis of optimal centroid estimation applied to Shack-Hartmann sensing," *Appl. Opt.*, vol. 38, no. 32, pp. 6737–6743, Nov 1999. DOI: <http://dx.doi.org/10.1364/AO.38.006737>
- [44] Z. Jiang, S. Gong, and Y. Dai, "Monte-Carlo analysis of centroid detected accuracy for wavefront sensor," *Optics & Laser Technology*, vol. 37, no. 7, pp. 541 – 546, 2005. DOI: <http://dx.doi.org/10.1016/j.optlastec.2004.08.009>
- [45] M. Kasper, E. Fedrigo, D. P. Looze, H. Bonnet, L. Ivanescu, and S. Oberti, "Fast calibration of high-order adaptive optics systems," *Journal of the Optical Society of America*, vol. 21, no. 6, pp. 1004–1008, Jun 2004. DOI: <http://dx.doi.org/10.1364/JOSAA.21.001004>
- [46] K. Kepa, D. Coburn, J. C. Dainty, and F. Morgan, "Re-baselining the ESO ELT project," *Analysis and roadmap from the ELT Adaptive Optics Working Group*, 2006.
- [47] K. Kepa, D. Coburn, J. C. Dainty, and F. Morgan, "High Speed Optical Wavefront Sensing with Low Cost FPGAs," *Measurement Science Review*, vol. 8, pp. 87–93, Jan. 2008. DOI: <http://dx.doi.org/10.2478/v10048-008-0021-z>
- [48] P. Y. Kern, P. J. Lena, P. Gigan, F. J. Rigaut, G. Rousset, J.-C. Fontanella, J.-P. Gaffard, C. Boyer, P. Jagourel, and F. Merkle, "Adaptive optics prototype system for infrared astronomy, i: system description," *Proceedings SPIE*, vol. 1271, pp. 243–251, 1990. DOI: <http://dx.doi.org/10.1117/12.20411>
- [49] H. K. Khalil, *Nonlinear Systems (3rd Edition)*, 3rd ed. Prentice Hall, 12 2001.
- [50] M. Knappek, "Adaptive Optics for the Mitigation of Atmospheric Effects in Laser Satellite-To-Ground Communications," *Dissertation, Technische Universität München*, 2011. [Online]. Available: <https://mediatum.ub.tum.de/?id=980518>
- [51] A. N. Kolmogorov, "The local structure of turbulence in incompressible viscous fluid for very large reynolds numbers," *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 434, no. 1890, pp. 9–13, 1991. DOI: <http://dx.doi.org/10.1098/rspa.1991.0075>
- [52] M. Le Louarn, N. Hubin, M. Sarazin, and A. Tokovinin, "New challenges for adaptive optics: extremely large telescopes," *Monthly Notices of the Royal Astronomical Society*, vol. 317, no. 3, pp. 535–544, 2000. DOI: <http://dx.doi.org/10.1046/j.1365-8711.2000.03607.x>
- [53] J. Lee, R. V. Shack, and M. R. Descour, "Sorting method to extend the dynamic range of the Shack-Hartmann wave-front sensor," *Appl. Opt.*, vol. 44, no. 23, pp. 4838–4845, Aug 2005. DOI: <http://dx.doi.org/10.1364/AO.44.004838>
- [54] G. R. Lemaître, *Astronomical Optics and Elasticity Theory: Active Optics Methods (Astronomy and Astrophysics Library)*, 1st ed. Springer, 1 2012.
- [55] N. Leonhard, R. Berlich, S. Minardi, A. Barth, S. Mauch, J. Mocci, M. Goy, M. Appelfelder, E. Beckert, and C. Reinlein, "Real-time adaptive optics testbed to investigate point-ahead angle in pre-compensation of earth-to-geo optical communication," *Opt. Express*, vol. 24, no. 12, pp. 13 157–13 172, Jun 2016. DOI: <http://dx.doi.org/10.1364/OE.24.013157>
- [56] C. Leroux and C. Dainty, "A simple and easy method to extend the dynamic range of an aberrometer," *Opt. Express*, vol. 17, no. 21, pp. 19 055–19 061, Oct 2009. DOI: <http://dx.doi.org/10.1364/OE.17.019055>

-
- [57] G. Leung, B. Francis, and J. Apkarian, “Bilateral controller for teleoperators with time delay via μ -synthesis,” *IEEE Transactions on Robotics and Automation*, vol. 11, no. 1, pp. 105–116, Feb 1995. DOI: <http://dx.doi.org/10.1109/70.345941>
 - [58] J. Liang, D. R. Williams, and D. T. Miller, “Supernormal vision and high-resolution retinal imaging through adaptive optics,” *Journal of the Optical Society of America*, vol. 14, no. 11, pp. 2884–2892, Nov 1997. DOI: <http://dx.doi.org/10.1364/JOSAA.14.002884>
 - [59] N. Lindlein, J. Pfund, and J. Schwider, “Expansion of the dynamic range of a Shack-Hartmann sensor by using astigmatic microlenses,” *Optical Engineering*, vol. 39, no. 8, pp. 2220–2225, 2000. DOI: <http://dx.doi.org/10.1117/1.1304846>
 - [60] N. Lindlein, J. Pfund, and J. Schwider, “Algorithm for expanding the dynamic range of a Shack-Hartmann sensor by using a spatial light modulator array,” *Optical Engineering*, vol. 40, no. 5, pp. 837–840, 2001. DOI: <http://dx.doi.org/10.1117/1.1357193>
 - [61] G. J. M., *Introduction to Wavefront Sensors (Tutorial Texts in Optical Engineering)*. Society of Photo Optical, 5 1995.
 - [62] P.-Y. Madec, “Overview of deformable mirror technologies for adaptive optics and astronomy,” *Proceedings SPIE*, vol. 8447, pp. 844 705–844 705–18, 2012. DOI: <http://dx.doi.org/10.1117/12.924892>
 - [63] J. D. Mansell, “Control Matrices Generation for Hartmann Wavefront Sensor Adaptive Optics,” *Homepage Active Optical Systems*, 2009. [Online]. Available: http://www.activeopticalsystems.com/docs/AN012_BasicsOfControlMatrices.pdf
 - [64] S. Mauch, A. Barth, J. Reger, N. Leonhard, and C. Reinlein, “Improved Thresholding and Ordering for Shack-Hartmann wavefront sensors implemented on an FPGA,” *International Workshop on Adaptive Optics for Industry and Medicine (AOIM)*, 2015.
 - [65] S. Mauch, A. Barth, J. Reger, C. Reinlein, M. Appelfelder, and E. Beckert, “FPGA-accelerated adaptive optics wavefront control part II,” *Proceedings SPIE*, vol. 9343, pp. 93 430Y–93 430Y–13, 2015. DOI: <http://dx.doi.org/10.1117/12.2079010>
 - [66] S. Mauch and J. Reger, “Application of μ -Synthesis H_∞ -Control for Adaptive Optics in Laser Material Processing,” in *2013 IEEE International Conference on Control Applications (CCA)*, 2013, pp. 941–947. DOI: <http://dx.doi.org/10.1109/CCA.2013.6662872>
 - [67] S. Mauch and J. Reger, “Real-time spot detection and ordering for a Shack-Hartmann wavefront sensor with a low-cost FPGA,” *IEEE Transactions on Instrumentation and Measurement*, 2014. DOI: <http://dx.doi.org/10.1109/TIM.2014.2310616>
 - [68] S. Mauch, J. Reger, and E. Beckert, “Adaptive Optics control for laser material processing,” *Congreso Latinoamericano de Control Automático (CLCA 2012)*, Peru, Lima, 2012. [Online]. Available: https://www.tu-ilmenau.de/fileadmin/public/regelungstechnik/Mitarbeiter/Prof_Johann_Reger/reger_CLCA2012.pdf

- [69] S. Mauch, J. Reger, C. Reinlein, M. Appelfelder, M. Goy, E. Beckert, and T. Tünnermann, “FPGA-accelerated adaptive optics wavefront control,” *Proceedings SPIE*, vol. 8978, pp. 1–12, 2014. DOI: <http://dx.doi.org/10.1117/12.2038910>
- [70] S. Mauch, “Neuartiges Verfahren basierend auf einem FPGA für die Auswertung eines Shack-Hartmann-Wellenfrontsensors,” german Patent 10 2014 017 050 A1, 12 18, 2013. [Online]. Available: <http://www.google.com/patents/DE102014017050A1?cl=de>
- [71] S. Mauch and J. Reger, “Real-Time Implementation of the Spiral Algorithm for Shack-Hartmann Wavefront Sensor Pattern Sorting on an FPGA,” *Measurement*, pp. –, 2016. DOI: <http://dx.doi.org/10.1016/j.measurement.2016.06.004>
- [72] R. K. MD, S. M. MD, and R. A. MD, Eds., *Wavefront Customized Visual Correction: The Quest for Super Vision II*, 2nd ed. Slack Incorporated, 11 2003.
- [73] S. Meimon, T. Fusco, and C. Petit, “The Slope-Oriented Hadamard scheme for in-lab or on-sky interaction matrix calibration,” in *Second International Conference on Adaptive Optics for Extremely Large Telescopes*, Sep. 2011. [Online]. Available: <http://adsabs.harvard.edu/abs/2011aoel.confE..65M>
- [74] G. ming (George) Dai, *Wavefront Optics for Vision Correction*. SPIE Publications, 2 2008. DOI: <http://dx.doi.org/10.1117/3.769212>
- [75] R. Navarro and J. Arines, “Complete Modal Representation with Discrete Zernike Polynomials - Critical Sampling in Non Redundant Grids,” *Numerical Simulations of Physical and Engineering Processes*, 2011. DOI: <http://dx.doi.org/10.5772/24631>
- [76] P.-B. Nguyen and S.-B. Choi, “Compensator design for hysteresis of a stacked pzt actuator using a congruency-based hysteresis model,” *Smart Materials and Structures*, vol. 21, no. 1, p. 015009, 2012. DOI: <http://dx.doi.org/10.1088/0964-1726/21/1/015009>
- [77] D. Noll, O. Prot, and P. Apkarian, “A proximity control algorithm to minimize nonsmooth and nonconvex functions,” *Journal of Convex Analysis*, vol. 13(3 & 4), pp. 641–666, 2009.
- [78] I. Optics, “Brochure HASO3 Fast SHWFS,” –, 2014. [Online]. Available: http://www.imagine-optic.com/downloads/brochures/imagine-optic_haso-fast_shack-hartmann_wavefront-sensor.pdf
- [79] M. Ortega and F. Rubio, “Systematic design of weighting matrices for the h_∞ mixed sensitivity problem,” *Journal of Process Control*, vol. 14, no. 1, pp. 89–98, feb 2004. DOI: [http://dx.doi.org/10.1016/s0959-1524\(03\)00035-0](http://dx.doi.org/10.1016/s0959-1524(03)00035-0)
- [80] M. Pachter and M. W. Oppenheimer, “Adaptive optics for airborne platforms—Part 1: modeling,” *Optics & Laser Technology*, vol. 34, no. 2, pp. 143–158, Mar. 2002. DOI: [http://dx.doi.org/10.1016/S0030-3992\(01\)00104-9](http://dx.doi.org/10.1016/S0030-3992(01)00104-9)
- [81] A. Packard, M. K. H. Fan, and J. Doyle, “A power method for the structured singular value,” in *Decision and Control, 1988., Proceedings of the 27th IEEE Conference on*, Dec 1988, pp. 2132–2137 vol.3. DOI: <http://dx.doi.org/10.1109/CDC.1988.194710>
- [82] S. I. Panagopoulou and D. R. Neal, “Zernike vs. zonal matrix iterative wavefront reconstructor,” –, 2005. [Online]. Available: <http://www.lumetrics.com/documents/wavefront/Zernike%20vs%20Zonal.pdf>

-
- [83] R. Paris, M. Thier, T. Thurner, and G. Schitter, "Shack-hartmann wavefront sensor based on an industrial smart camera," in *2012 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, May 2012, pp. 1127–1132. DOI: <http://dx.doi.org/10.1109/I2MTC.2012.6229194>
- [84] B. C. Platt and R. Shack, "History and principles of Shack-Hartmann wavefront sensing." *Journal of refractive surgery*, vol. 17, no. 5, 2001.
- [85] Polytec, "PSV-400 Scanning Vibrometer," -, 2014. [Online]. Available: http://www.polytec.com/fileadmin/user_uploads/Products/Vibrometers/PSV-400/Documents/OM_DS_PSV-400_2011_05_E.pdf
- [86] L. A. Poyneer and J.-P. Véran, "Optimal modal fourier-transform wavefront control," *Journal of the Optical Society of America*, vol. 22, no. 8, pp. 1515–1526, Aug 2005. DOI: <http://dx.doi.org/10.1364/JOSAA.22.001515>
- [87] I. Prempain, Emmanuel; Postlethwaite, "A Constant D-Scale μ -Synthesis Approach Based on Nonsmooth Optimization," in *Proceedings of the 17th IFAC World Congress, 2008*, vol. 17. IFAC-PapersOnLine, 2008. DOI: <http://dx.doi.org/10.3182/20080706-5-KR-1001.02572>
- [88] R. Ragazzoni, "Pupil plane wavefront sensing with an oscillating prism," *Journal of Modern Optics*, vol. 43, no. 2, pp. 289–293, 1996. DOI: <http://dx.doi.org/10.1080/09500349608232742>
- [89] R. Ragazzoni, E. Diolaiti, and E. Vernet, "A pyramid wavefront sensor with no dynamic modulation," *Optics Communications*, vol. 208, no. 1-3, pp. 51 – 60, 2002. DOI: [http://dx.doi.org/10.1016/S0030-4018\(02\)01580-8](http://dx.doi.org/10.1016/S0030-4018(02)01580-8)
- [90] C. Reinlein, "Thermo-mechanical design, realization and testing of screen-printed deformable mirrors," Ph.D. dissertation, Technische Universität Ilmenau, 2012. [Online]. Available: <http://www.db-thueringen.de/servlets/DocumentServlet?id=20398>
- [91] C. Reinlein, M. Appelfelder, S. Gebhardt, E. Beckert, R. Eberhardt, and A. Tünnermann, "Thermomechanical design, hybrid fabrication, and testing of a MOEMS deformable mirror," *Journal of Micro/Nanolithography, MEMS, and MOEMS*, vol. 12, no. 1, pp. 013 016–013 016, 2013. DOI: <http://dx.doi.org/10.1117/1.JMM.12.1.013016>
- [92] C. Reinlein, M. Appelfelder, M. Goy, K. Ludewigt, and A. Tünnermann, "Performance of a thermal-piezoelectric deformable mirror under 6.2 kW continuous-wave operation," *Applied Optics*, vol. 52, no. 34, pp. 8363–8368, Dec 2013. DOI: <http://dx.doi.org/10.1364/AO.52.008363>
- [93] M. Rezac and Z. Hurak, "Structured mimo design for dual-stage inertial stabilization: Case study for hifoo and hinfstruct solvers," *Mechatronics*, vol. 23, no. 8, pp. 1084–1093, 2013. DOI: <http://dx.doi.org/10.1016/j.mechatronics.2013.08.003>
- [94] J. C. Ricklin and F. M. Davidson, "Atmospheric turbulence effects on a partially coherent gaussian beam: implications for free-space laser communication," *Journal of the Optical Society of America*, vol. 19, no. 9, pp. 1794–1802, Sep 2002. DOI: <http://dx.doi.org/10.1364/JOSAA.19.001794>
- [95] F. Roddier, Ed., *Adaptive Optics in Astronomy*. Cambridge University Press, 11 2004.

- [96] F. Roddier and C. Roddier, "Wavefront reconstruction using iterative fourier transforms," *Applied Optics*, vol. 30, no. 11, pp. 1325–1327, Apr 1991. DOI: <http://dx.doi.org/10.1364/AO.30.001325>
- [97] A. Roorda, F. Romero-Borja, I. William Donnelly, H. Queener, T. Hebert, and M. Campbell, "Adaptive optics scanning laser ophthalmoscopy," *Optics Express*, vol. 10, no. 9, pp. 405–412, May 2002. DOI: <http://dx.doi.org/10.1364/OE.10.000405>
- [98] S. Esposito and A. Riccardi, "Pyramid wavefront sensor behavior in partial correction adaptive optic systems," *Astronomy & Astrophysics*, vol. 369, no. 2, pp. L9–L12, 2001. DOI: <http://dx.doi.org/10.1051/0004-6361:20010219>
- [99] C. Scherer, "Theory of robust control," *Slide Course Notes, Delft University of Technology*, 2005. [Online]. Available: <http://www.dcsc.tudelft.nl/~cscherer/robust/Slides05.pdf>
- [100] C. Scherer, "Theory of robust control," *Course Notes, Delft University of Technology*, 2007.
- [101] J. Schwiegerling, "Description of zernike polynomials," -, Feb 2011. [Online]. Available: <http://www.visualopticslab.com/OPTI515L/Background/Zernike%20Notes%2017Feb2011.pdf>
- [102] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, 2nd ed. Wiley-Interscience, 11 2005.
- [103] D. G. Smith, *High dynamic range calibration for an infrared Shack-Hartmann wavefront sensor*. ProQuest, UMI Dissertation Publishing, 9 2011. [Online]. Available: http://arizona.openrepository.com/arizona/bitstream/10150/194779/1/azu_etd_2539_sip1_m.pdf
- [104] D. G. Smith and J. E. Greivenkamp, "Generalized method for sorting Shack-Hartmann spot patterns using local similarity," *Applied Optics*, vol. 47, no. 25, pp. 4548–4554, Sep 2008. DOI: <http://dx.doi.org/10.1364/AO.47.004548>
- [105] W. H. Southwell, "Wave-front estimation from wave-front slope measurements," *Journal of the Optical Society of America*, vol. 70, no. 8, pp. 998–1006, Aug 1980. DOI: <http://dx.doi.org/10.1364/JOSA.70.000998>
- [106] Y. Z. Tan, C. K. Pang, F. Hong, S. Won, and T. H. Lee, "Hysteresis compensation of piezoelectric actuators in dual-stage hard disk drives," in *2011 8th Asian Control Conference (ASCC)*, May 2011, pp. 1024–1029.
- [107] V. I. Tatarskii, *Wave Propagation in Turbulent Medium*. McGraw-Hill, 1961.
- [108] S. Thomas, T. Fusco, A. Tokovinin, M. Nicolle, V. Michau, and G. Rousset, "Comparison of centroid computation algorithms in a Shack-Hartmann sensor," *Monthly Notices of the Royal Astronomical Society*, vol. 371, no. 1, pp. 323–336, 2006. DOI: <http://dx.doi.org/10.1111/j.1365-2966.2006.10661.x>
- [109] T. Trenz. (2013, Dec.) trenz electronic. [Online]. Available: <http://www.trenz-electronic.de>
- [110] Truesense, "Kai-0340 image sensor," -, Jan 2015. [Online]. Available: http://www.ccd.com/pdf/ccd_340.pdf
- [111] R. Tyson, *Principles of Adaptive Optics, Third Edition (Series in Optics and Optoelectronics)*, 3rd ed. CRC Press, 9 2010.

-
- [112] R. K. Tyson, "Bit-error rate for free-space adaptive optics laser communications," *Journal of the Optical Society of America*, vol. 19, no. 4, pp. 753–758, Apr 2002. DOI: <http://dx.doi.org/10.1364/JOSAA.19.000753>
 - [113] R. K. Tyson, *Principles of Adaptive Optics, Fourth Edition*, 4th ed. CRC Press, 11 2015.
 - [114] R. K. Tyson and B. W. Frazier, *Field Guide to Adaptive Optics, 2nd Ed (SPIE Field Guide Vol. FG24)*, 2nd ed. SPIE Press, 4 2012.
 - [115] P. Wizinowich, "Adaptive optics and keck observatory," *IEEE Instrumentation Measurement Magazine*, vol. 8, no. 2, pp. 12–19, Jun 2005. DOI: <http://dx.doi.org/10.1109/MIM.2005.1405918>
 - [116] M. Xia, C. Li, L. Hu, Z. Cao, Q. Mu, and L. Xuan, "Shack-hartmann wavefront sensor with large dynamic range," *Journal of Biomedical Optics*, vol. 15, no. 2, pp. 026 009–026 009–10, 2010. DOI: <http://dx.doi.org/10.1117/1.3369810>
 - [117] X. Yin, X. Li, L. Zhao, and Z. Fang, "Adaptive thresholding and dynamic windowing method for automatic centroid detection of digital shack-hartmann wavefront sensor," *Applied Optics*, vol. 48, no. 32, pp. 6088–6098, Nov 2009. DOI: <http://dx.doi.org/10.1364/AO.48.006088>
 - [118] X. Yin, X. Li, L. Zhao, and Z. Fang, "Adaptive thresholding and dynamic windowing method for automatic centroid detection of digital Shack-Hartmann wavefront sensor," *Appl. Opt.*, vol. 48, no. 32, pp. 6088–6098, 2009. DOI: <http://dx.doi.org/10.1364/AO.48.006088>
 - [119] G. Yoon, S. Pantanelli, and L. J. Nagy, "Large-dynamic-range shack-hartmann wavefront sensor for highly aberrated eyes," *Journal of Biomedical Optics*, vol. 11, no. 3, pp. 030 502–030 502–3, 2006. DOI: <http://dx.doi.org/10.1117/1.2197860>
 - [120] G. Zames, "On the input-output stability of time-varying nonlinear feedback systems—part ii: Conditions involving circles in the frequency plane and sector nonlinearities," *IEEE Transactions on Automatic Control*, vol. 11, no. 3, pp. 465–476, Jul 1966. DOI: <http://dx.doi.org/10.1109/TAC.1966.1098356>
 - [121] G. Zames, "On the input-output stability of time-varying nonlinear feedback systems part one: Conditions derived using concepts of loop gain, conicity, and positivity," *IEEE Transactions on Automatic Control*, vol. 11, no. 2, pp. 228–238, Apr 1966. DOI: <http://dx.doi.org/10.1109/TAC.1966.1098316>
 - [122] H. Zeller, "Cosmic ray induced failures in high power semiconductor devices," *Solid-State Electronics*, vol. 38, no. 12, pp. 2041 – 2046, 1995. DOI: [http://dx.doi.org/10.1016/0038-1101\(95\)00082-5](http://dx.doi.org/10.1016/0038-1101(95)00082-5)
 - [123] A. Zepp, S. Gładysz, and K. Stein, "Holographic wavefront sensor for fast defocus measurement," *Advanced Optical Technologies*, vol. 2, no. 5-6, 2013. DOI: <http://dx.doi.org/10.1515/aot-2013-0050>
 - [124] K. Zhou and J. C. Doyle, *Essentials of Robust Control*, 1st ed. Prentice Hall, 10 1997.
 - [125] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*, 1st ed. Prentice Hall, 8 1995.
 - [126] X. Zhu and J. Kahn, "Free-space optical communication through atmospheric turbulence channels," *IEEE Transactions on Communications*, vol. 50, no. 8, pp. 1293–1300, Aug 2002. DOI: <http://dx.doi.org/10.1109/TCOMM.2002.800829>