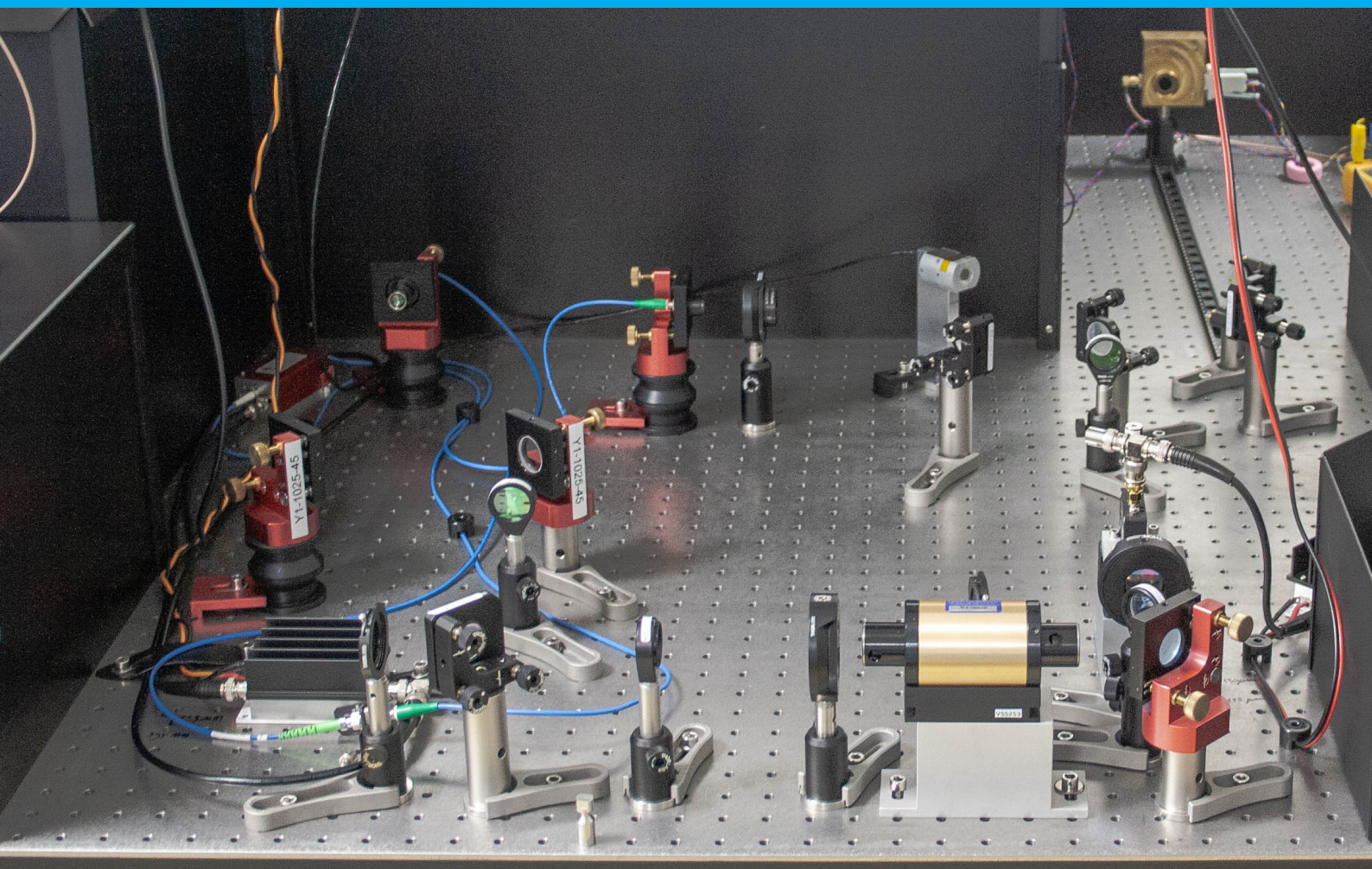
Characterization of a phase camera for the Virgo gravitational wave detector

F. J. Hoogenberg

Master thesis
MSc Applied Physics



Nikhef TU Delft

Characterization of a phase camera for the Virgo gravitational wave detector

by

F. J. Hoogenberg

to obtain the degree of

Master of Science
in Applied Physics
Physics for Instrumentation track

in the Faculty of Applied Sciences
at the Delft University of Technology,
to be defended publicly on Tuesday, June 20, 2023, at 10:00 A.M.
in the Mooijzaal, building 22 (Applied Physics) room F.005.

student number:	4551664	
project duration:	September 19, 2022 – June 20, 2023	
thesis committee:	dr. M. Fransen	Nikhef, responsible supervisor
	dr. ir. S.F. Pereira	TU Delft, committee member
	dr. B. Rieger	TU Delft, committee member
	dr. ing. M.G. van Beuzekom	Nikhef, daily supervisor

An electronic version of this thesis is available at https://repository.tudelft.nl/.

TU Delft Nikhef

Preface

Before you lies the master thesis ”Characterization of a phase camera for the Virgo gravitational wave detector”. I have written this thesis to fulfill the graduation requirements of the Applied Physics program at the Delft University of Technology. I was engaged in researching and writing this thesis at Nikhef in Amsterdam from September 2022 till June 2023.

During my studies, I noticed that I was most interested the new and groundbreaking physics. Courses such as General Relativity and Introduction to Elementary Particle Physics were most appealing to me. Via this last course, taught by Ernst-Jan Buis and Martin Fransen, I came into contact with the National Institute for Subatomic Physics in Amsterdam, also called Nikhef. After some talks, I met Niels van Bakel, head of the detector R&D group at Nikhef. He showed me several projects with the phase camera project the most interesting due to the combination of simulations and experiments in the laboratory.

Now almost a year and many train rides later, I can finally show what I have accomplished and learned during this thesis. In the lab, I learned to build optical set-ups and trained my patience and precision. Behind my desk, the data analysis and fundamental discussions about what happens in the set-up took place. During all of this, I have learned that almost nothing goes according to plan in new research. Which makes the rewards of success even better.

Of course, I could not have done this project without the help I have received. First of all, I wish to thank Martin van Beuzekom, my daily supervisor from Nikhef, for all the knowledge he shared with me and for always being present, online or in person and during the day and night, for questions and support. Next, I would like to thank Martin Fransen, my responsible supervisor, for making this project possible and enthusing me about Nikhef. I also want to thank the Nikhef institute for the opportunities they gave me to have a look into the academic world such as during congresses. Besides the project-related support, I would like to thank the detector R&D group for the meaningless discussions during the breaks. And finally, I wish to thank Silvania Pereira and Bernd Rieger for completing my assessment committee and for their independent judgment on my work.

This thesis marks the end of my time as a student. I have had a great and fun student time during which I made many long-lasting friendships. Thank you all for the pleasure and support over the last few years. Now nothing remains, but to thank you as my reader and to hope you enjoy your reading.

F. J. Hoogenberg
Amsterdam, June 9, 2023

Abstract

Gravitational waves predicted by Einstein have been measured in a few observatories worldwide. These detectors are complex interferometers with sizes of kilometers. Misalignments and aberrations in their optical set-up can create higher-order modes in the laser beam. These higher-order mode effects should be compensated to improve the sensitivity of the interferometer. Such a compensating control system requires a real-time wavefront sensor. The phase cameras developed for the Virgo gravitational wave detector can simultaneously create intensity and phase images of the laser wavefront at 11 demodulation frequencies. Currently, the phase images are not used due to difficulties in the interpretation. Therefore, the goal of this research is to improve the understanding of the phase images of a phase camera for the Virgo gravitational wave detector.

A prototype set-up of the phase camera was built at Nikhef. The beam is modulated and sidebands are created such that a beat signal of 80 MHz and the first upper and lower sidebands at 75 and 85 MHz are measured. Images are created by scanning the laser beam across a pinhole diode and digital demodulation. In the chosen optical set-up only one of the two beams that give the measurable beat signal is scanned. This leads to systematic phase effects caused by length differences in the optical path as a function of the scanning angle. To predict the phase images, the interference of two Gaussian beams with a scanning mirror is mathematically derived. Phase images are created with a simulation, which matches within 10 to 20% compared to measurements. To improve the results, the input parameters of the simulation should be measured more accurately and the optical set-up layout should be checked. Also, the phase stability of the prototype set-up is measured. Due to the high sensitivity of the phase camera, very small optical path length differences, for example caused by airflow or temperature differences, lead to significant phase offsets. The long timescale fluctuations in the phase measurements can be reduced by covering the optical set-up. The measured high-frequency phase resolution is $\Delta\phi = 7.1 \pm 0.4$ mrad and behaves as a function of power as expected.

Table of contents

Preface iii

Abstract v

1 Introduction 1

2 Theory 3
2.1 Gaussian beam optics . . . 3
2.1.1 Paraxial Helmholtz equation . . . 3
2.1.2 Gaussian beam propagation . . . 4
2.1.3 Real beam propagation . . . 5
2.1.4 Higher-order modes . . . 7
2.1.4.1 Hermite-Gaussian modes . . . 7
2.1.5 Transformation and magnification by simple lenses . . . 8
2.2 Sidebands . . . 10
2.3 Beam profiling . . . 11

3 Phase camera 13
3.1 Working principle of the phase camera . . . 13
3.2 Optical set-up . . . 14
3.2.1 Acousto-optic modulator . . . 15
3.2.2 Electro-optic modulator . . . 16
3.2.3 Scanner . . . 18
3.2.4 Photodiode . . . 20
3.2.5 Demodulation . . . 20
3.3 Beam conditioning . . . 22

4 Phase imaging 25
4.1 Interference of two plane waves . . . 25
4.1.1 Homodyne plane waves . . . 25
4.1.2 Heterodyne plane waves . . . 26
4.2 Interference of two Gaussian beams . . . 26
4.2.1 Curved wavefront . . . 27
4.2.2 Gouy phase . . . 28
4.3 Interference with a scanning mirror . . . 28
4.4 Interference in the phase camera set-up . . . 33
4.4.1 Total phase . . . 33
4.4.2 Sidebands . . . 33
4.5 Simulation of the phase images . . . 34
4.6 Validation of the simulation . . . 36
4.6.1 Single measurement . . . 36
4.6.2 Measurement method for the number of phase rings . . . 37
4.6.3 Number of phase rings versus photodiode position . . . 38

5 Phase stability 41
5.1 Long timescale phase stability . . . 41
5.2 Phase resolution . . . 42
5.2.1 Measurement method . . . 43
5.2.2 Power-dependent measurements . . . 44

6 Discussion **49**
6.1 Phase centering 49
6.2 Phase image simulation 49
6.3 Phase stability 50

7 Summary & conclusion **51**

Bibliography **54**

Appendices **55**

A Laguerre-Gaussian modes **57**

B Python scripts **59**
B.1 Data processing 59
B.2 Phase image simulation 65

C Simulation validation measurements **71**

1

Introduction

In November 1915, Albert Einstein published his general theory of relativity (Einstein, 1915a, 1915b, 1915c, 1915d). His theory refines Newton's law of universal gravitation and describes gravity as a geometric property of space and time. This description of gravity led to the prediction of the existence of gravitational waves. Einstein himself doubted whether these faint and weakly interacting waves could ever be detected (Cervantes-Cota et al., 2016). However, exactly 100 years later on September 14, 2015, the first direct observation of a gravitational wave was made (Abbott et al., 2016). The waveform was detected by both LIGO observatories in the United States, which are two of the few gravitational wave observatories. In Europe, the Virgo gravitational wave detector is located in Cascina in Italy.

These three detectors use a laser Michelson interferometer to measure the gravitational waves induced length contraction (Weiss, 1972). The interferometers have two arms of multiple kilometers that are orthogonal to each other. A laser beam is split at the crossing point of the two arms and sent into both of them. The light is reflected at the end of the arms and recombines again at the beam splitter at the crossing point. If both arms have a length that is exactly a multiple of the wavelength of the laser light, then the beams are in phase and interfere destructively at the beam splitter. However, a passing gravitational wave stretches one of the arms, while at the same time contracts the other arm. This tiny difference in length causes the interference to be not fully destructive and hence light appears at the so-called dark port.

An interferometer can measure very small length differences between both arms. This high sensitivity makes gravitational wave detection possible. However, it makes them also susceptible to distortions. In the optical set-up of an interferometer, aberrations or misalignments can create higher-order modes in the laser beam. Aberrations can be due to imperfections in the optics and due to thermal effects (Van der Schaaf, 2020). These higher-order modes reduce the signal power in the cavities resulting in a lower sensitivity of the gravitational wave detector.

To improve the sensitivity of a gravitational wave detector, the power of the laser beams is increased. This leads to more thermally induced aberrations. To compensate for the induced higher-order modes, a compensating control system with adaptive optics is added. For this control system, a wavefront sensor is needed to detect the higher-order modes. A phase camera is an ideal candidate since it can simultaneously measure the intensity and phase of the laser wavefront at multiple frequencies. Hence the role of phase cameras will become only more important (Acernese et al., 2014).

To control the interferometer, modulated laser beams are used. The phase camera uses the same modulation frequencies and uses a heterodyne demodulation schema to independently probe the upper and lower sideband of each modulation frequency. The frequencies are chosen so that they only resonate in the least possible cavities such that each sideband only probe those cavities for high-order modes (Agatsuma et al., 2019).

Phase cameras have already been used in the LIGO detectors, but only the amplitude information was used because of difficulties in the interpretation of the phase information (Gretarsson et al., 2007). In the Advanced Virgo detector, the phase camera is employed as part of a thermal compensation system to mitigate aberration effects which are due to the heating of the mirrors by the laser beams (Rocchi et al., 2012). Besides, the relative phase shift between the carrier and sidebands can be used to analyze the state of a cavity according to Day (2013). Therefore to improve the current gravitational wave detectors, a detailed understanding of phase images of a phase camera is needed.

In this research, a prototype set-up of the Virgo phase camera was built at Nikhef. This prototype set-up is used to characterize the phase camera and aid in the understanding of the phase images. Systematic effects of scanning on the phase images were mathematically derived and compared with measurements. Besides, the stability of the phase images is determined and possible improvements are discussed.

Before the prototype set-up is discussed in this report, chapter 2 introduces the required theoretical background on Gaussian laser beams. The optical set-up and calibration of the prototype set-up are discussed in chapter 3. Subsequently, chapter 4 describes a mathematical derivation and a simulation of the systematic effects of scanning on the phase images, which are compared with measurements. In chapter 5, the phase stability of the prototype set-up is measured and results are discussed. Finally, chapters 6 and 7 are a discussion and conclusion on the results of this work, respectively.

2

Theory

The phase camera is an instrument that measures the wavefront of a laser beam. To build and understand this instrument some theoretical knowledge is required. First, Gaussian beam optics is described to understand the behavior of laser beams. Next, sidebands are explained, which are used in the phase camera set-up. Finally, a measurement method is explained to extract the properties of a laser beam.

2.1. Gaussian beam optics

A laser beam can be described as a Gaussian beam. This type of beam has special transformation properties. This section focuses on the behavior and properties of Gaussian beams, starting with a derivation from the Maxwell equations.

2.1.1. Paraxial Helmholtz equation

The emitted light of a laser can be described with the Maxwell equations in vacuum

$$\begin{aligned} \vec{\nabla} \cdot \vec{E} &= 0, \qquad \vec{\nabla} \times \vec{E} = -\frac{\partial \vec{B}}{\partial t}, \\ \vec{\nabla} \cdot \vec{B} &= 0, \qquad \vec{\nabla} \times \vec{B} = \epsilon_0 \mu_0 \frac{\partial \vec{E}}{\partial t}, \end{aligned} \tag{2.1}$$

where $\vec{E}$ is the electric field, $\vec{B}$ the magnetic field, ϵ_0 the vacuum permittivity and μ_0 the vacuum permeability. Taking the curl of the Maxwell-Faraday equation (right-top) and swapping the linear operators of the magnetic field results in

$$\vec{\nabla} \times \left(\vec{\nabla} \times \vec{E} \right) = -\frac{\partial}{\partial t} \left(\vec{\nabla} \times \vec{B} \right). \tag{2.2}$$

Substituting Gauss's and Ampère's law into this equation returns the wave equation

$$\nabla^2 \vec{E} = \epsilon_0 \mu_0 \frac{\partial^2 \vec{E}}{\partial t^2}. \tag{2.3}$$

This partial differential equation can be solved with the method of separation of variables. Therefore, it is assumed that $\vec{E}\,(\vec{x}, t) = \vec{X}\,(\vec{x}) \cdot \vec{T}\,(t)$. Substituting the assumption into the wave equation and rewriting it gives a constant

$$\frac{\nabla^2 \vec{X}}{X^2} \cdot \vec{X} = \epsilon_0 \mu_0 \frac{\partial^2 \vec{T}}{T \partial t^2} \equiv -k^2, \tag{2.4}$$

which is defined as $-k^2$. Note that the solution of the temporal part equals

$$T\,(t) = e^{i\omega t}, \tag{2.5}$$

where $\omega = \frac{k}{\sqrt{\epsilon_0 \mu_0}} = kc$ is the angular frequency, c the speed of light and k becomes the wave number. The spatial differential equation is called the Helmholtz equation (Saleh & Teich, 2009)

$$\nabla^2 \vec{X} + k^2 \vec{X} = 0. \tag{2.6}$$

Now assume the solution is a wave propagating in the z-direction, so let $\vec{X}(\vec{x}) = \vec{U}(\vec{x}) \cdot e^{-ikz}$. Substituting this solution into the Helmholtz equation results in

$$\nabla^2\vec{X} + k^2\vec{X} = \left(\partial_x^2\vec{U} + \partial_y^2\vec{U} + \partial_z^2\vec{U} - 2ik\partial_z\vec{U}\right) \cdot e^{-ikz} = 0. \tag{2.7}$$

Note that part of the Laplacian cancels out the term $k^2\vec{X}$. Next, assume a slowly varying envelope or paraxial approximation, $\left|\partial_z^2\vec{U}\right| \ll \left|2ik\partial_z\vec{U}\right|$. This is valid for waves of which the envelope varies slowly compared to the wavelength, as is the case for laser light. Using this approximation, the paraxial Helmholtz equation is derived (Saleh & Teich, 2009)

$$\nabla_\perp^2\vec{U} - 2ik\partial_z\vec{U} = 0, \tag{2.8}$$

where $\nabla_\perp^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is the transverse part of the Laplace operator.

A group of solutions to the paraxial Helmholtz equation are the so-called Gaussian beams. First, the lowest-order mode will be described. Later in section 2.1.4, higher-order modes will be discussed. The field amplitude in free space of the lowest-order mode equals (Svelto & Hanna, 2010)

$$\vec{U}(x,y,z) = \frac{w_0}{w(z)} e^{-\frac{x^2+y^2}{w(z)^2}} e^{-i\left(k\frac{x^2+y^2}{2R(z)} - \Phi\right)}, \tag{2.9}$$

where w_0 is the size of the beam waist, $w(z)$ the beam radius at position z, $R(z)$ the radius of curvature of the wavefront and $\Phi = \arctan\left(\frac{z}{z_R}\right)$ the Gouy phase for the lowest-order mode. These parameters are discussed in the next section.

Combining the field amplitude, with the assumption that the wave is traveling in the z-direction, and the temporal solution, equation 2.5, the complex electric field of a lowest-order Gaussian beam is found

$$\vec{E}(t,x,y,z) = E_0 \frac{w_0}{w(z)} e^{-\frac{x^2+y^2}{w(z)^2}} e^{-i\left(kz + k\frac{x^2+y^2}{2R(z)} - \Phi\right)} e^{i\omega t}\hat{x}, \tag{2.10}$$

with E_0 the electric field amplitude at the origin and $\hat{x}$ the polarization vector. Note, this is a theoretical TEM_{00} mode, meaning that this is a lowest-order mode of electromagnetic radiation where the electric and the magnetic field are both perpendicular to the direction of propagation (Zangwill, 2012).

2.1.2. Gaussian beam propagation

The intensity of the lowest-order mode of a Gaussian beam is described by a Gaussian distribution, which explains its name,

$$I(x,y,z) = \left|\vec{E}(t,x,y,z)\right|^2 = \frac{2P}{\pi w(z)^2} e^{\frac{-2\left(x^2+y^2\right)}{w(z)^2}}, \tag{2.11}$$

with P being the total power of the beam. Since theoretically the intensity of a Gaussian beam extends infinitely, a definition has to be chosen. In this thesis, the diameter of a Gaussian beam is defined as the width at which the beam intensity has fallen to $1/e^2$ of its maximum (CVI Melles Griot, n.d.). Due to diffraction, this diameter is not constant while propagating and can be described with

$$w(z) = w_0\sqrt{1 + \left(\frac{\lambda z}{\pi w_0^2}\right)^2}, \tag{2.12}$$

where w is the beam radius at which the intensity falls to $1/e^2$ of its maximum, w_0 the beam waist, λ the wavelength and z the distance from the waist. Besides the beam radius increasing when moving away from the beam waist, the wavefront of a Gaussian beam curves while propagating due to diffraction. The radius of curvature (RoC) of the wavefront is given by

$$R(z) = z\left[1 + \left(\frac{\pi w_0^2}{\lambda z}\right)^2\right]. \tag{2.13}$$

The wavefront is flat at the smallest radius of the beam, the beam waist, position $z_0 = 0$ and for $z \to \pm\infty$. Note, a flat wavefront implies a surface with a constant phase. For $z \to \pm\infty$, the beam radius asymptotically approaches a cone with a radius of $\theta = \frac{w(z)}{z} = \frac{\lambda}{\pi w_0}$. In figure 2.1, the wavefront curvature and the asymptotic cone can be seen. Starting from the waist, the curvature first increases up to a maximum and then decreases again. The maximum curvature is at the Rayleigh range z_R, which is defined as the position where the beam radius equals $w(z_R) = \sqrt{2}w_0$. This results in

$$z_R = \frac{\pi w_0^2}{\lambda}. \tag{2.14}$$

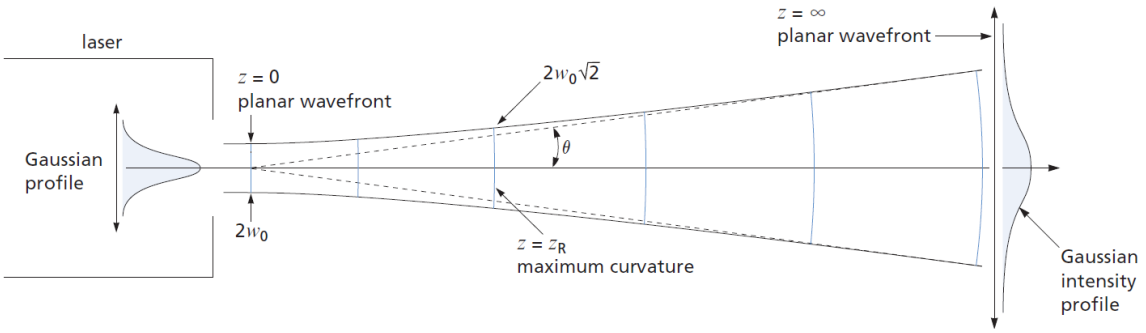
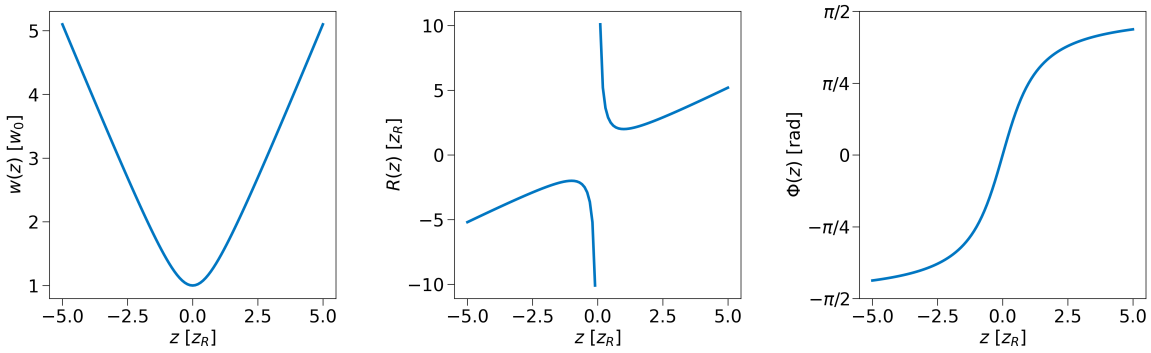


Figure 2.1: Gaussian beam emitted from a laser whose beam radius and wavefront radius of curvature evolve while propagating. Adapted from CVI Melles Griot (n.d.).

The parameters to describe a Gaussian beam are the beam radius, wavefront RoC and Gouy phase. The latter is an extra longitudinal phase lag originating from the slower phase velocity of a Gaussian beam compared to a plane wave (Freise & Strain, 2010). These three quantities all vary along the direction of propagation as derived above. For completeness, the normalized evolution of these quantities is shown in figure 2.2.



(a) beam radius (b) wavefront radius of curvature (c) Gouy phase

Figure 2.2: Dependence of the beam radius (a), wavefront radius of curvature (b) and Gouy phase (c) of a Gaussian beam on the position along the direction of propagation.

2.1.3. Real beam propagation

In the previous two sections, the lowest-order Gaussian beam has been theoretically derived. However, in the real world, it is difficult to create a perfectly Gaussian beam. To describe the deviations from the fundamental Gaussian beam, a dimensionless beam propagation parameter, $w_0\theta$, is used. This beam parameter stays constant throughout optical systems (CVI Melles Griot, n.d.). For theoretical Gaussian beams, this parameter equals $\frac{\lambda}{\pi}$. The ratio of the beam propagation parameter of a real beam and a

theoretical Gaussian beam is defined as the beam quality factor M^2

$$M^2 = \frac{w_{0R}\theta_R}{w_0\theta}, \tag{2.15}$$

where w_{0R} and θ_R are the beam waist and far-field divergence half angle of a real beam. Table 2.1 shows typical quality factors for different types of laser beams. The used laser beam in this research has a quality factor of approximately $M^2 \approx 1.01$.

Table 2.1: Typical quality factors for different types of laser beams (CVI Melles Griot, n.d.).

laser type	M^2
theoretical Gaussian beam	1
helium-neon laser in TEM_{00} mode	<1.1
ion lasers	1.1-1.7
high-energy multimode lasers	>10

The waist and the far-field divergence of a mixed-mode Gaussian beam are M times larger than the theoretical beam. Figure 2.3 shows a theoretical Gaussian beam and a realistic beam, as well as their parameters.

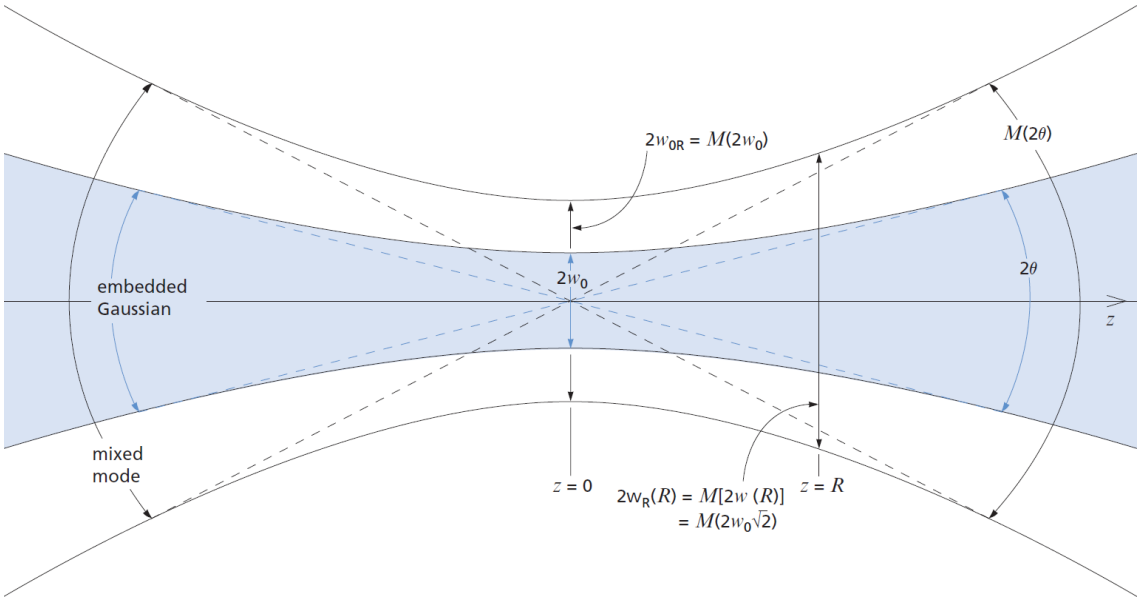


Figure 2.3: Embedded Gaussian beam and a realistic Gaussian beam. Note that the beam waist and far-field divergence scale with M, while the wavefront curvature and Rayleigh range are independent of M. Adapted from CVI Melles Griot (n.d.).

For a real Gaussian beam the propagation equations (beam radius, wavefront curvature and Rayleigh range) alter from their theoretical equations 2.12 to 2.14. The beam radius is now written as

$$w_R(z) = w_{0R}\sqrt{1 + \left(\frac{\lambda z M^2}{\pi w_{0R}^2}\right)^2}, \tag{2.16}$$

where the subscript R denotes the real beam. The equation for the real wavefront radius of curvature becomes

$$R_R(z) = z\left[1 + \left(\frac{\pi w_{0R}^2}{\lambda z M^2}\right)^2\right]. \tag{2.17}$$

Since the waist size and divergence both scale with M, the Rayleigh range remains the same as that

of a theoretical beam (equation 2.14). It can be expressed in terms of the real beam waist as

$$z_R = \frac{\pi w_{0R}^2}{M^2 \lambda} = \frac{\pi \left(M w_0\right)^2}{M^2 \lambda} = \frac{\pi w_0^2}{\lambda}. \tag{2.18}$$

2.1.4. Higher-order modes

The paraxial Helmholtz equation was derived in section 2.1.1. Its lowest-order solution equals the fundamental Gaussian TEM_{00} mode (equation 2.10). If a laser emits this fundamental mode, unwanted higher-order modes may be created in the optical set-up. These modes could for example originate from misalignments, mode mismatching of cavities, (surface) defects of elements and thermal lensing effects (Saleh & Teich, 2009). The phase camera can measure higher-order modes and can be used to correct for them (Van der Schaaf, 2020). In this section, higher-order modes are described. In particular, a family of orthogonal modes is discussed, Hermite-Gaussian (HG) modes. These correspond to rectangular symmetric modes. Alternatively, other families of orthogonal modes could be used such as the Laguerre-Gaussian (LG) modes, which are discussed in appendix A (Kogelnik & Li, 1966).

2.1.4.1. Hermite-Gaussian modes

The Hermite-Gaussian (HG) modes are described in the Cartesian coordinate system and form a complete set. Therefore, the electric field of a coherent paraxial beam can be written as a linear combination of Hermite-Gaussian modes

$$\vec{E}\left(x,y,z\right) = \sum_{lm} A_{lm} U_{lm}\left(x,y,z\right) e^{-ikz+i\omega t} \vec{e}, \tag{2.19}$$

where A_{lm} is the complex amplitude for each mode U_{lm} and $\vec{e}$ is the polarization of the field. Note, the lowest mode is defined as U_{00} since $l, m \in \mathbb{N}$. The modes are normalized according to

$$\int U_{kl}\left(x,y,z\right) U_{mn}^{*}\left(x,y,z\right) dxdy = \begin{cases} 1 & \text{for } k = m \text{ and } l = n \\ 0 & \text{else} \end{cases}. \tag{2.20}$$

The power in a mode is defined as the complex amplitude squared $|A_{lm}|^2$. The Hermite-Gaussian modes are given by (Bond et al., 2016)

$$U_{lm}\left(x,y,z\right) = \sqrt{\frac{1}{2^{l+m-1} l! m! \pi}} \frac{1}{w\left(z\right)} H_l\left(\frac{\sqrt{2}x}{w\left(z\right)}\right) H_m\left(\frac{\sqrt{2}y}{w\left(z\right)}\right) e^{-\frac{x^2+y^2}{w(z)^2}} e^{-i\frac{k\left(x^2+y^2\right)}{2R(z)} + i(l+m+1)\Phi(z)}, \tag{2.21}$$

where H_l and H_m are Hermite polynomials. The first few Hermite polynomials equal

$$H_0\left(x\right) = 1, \quad H_1\left(x\right) = 2x, \quad H_2\left(x\right) = 4x^2 - 2 \quad \text{and} \quad H_3\left(x\right) = 8x^3 - 12. \tag{2.22}$$

In figure 2.4 the amplitude and phase of the fundamental mode $U_{00}\left(x,y,z\right)$ and first higher-order HG mode $U_{10}\left(x,y,z\right)$ are shown for a beam with $\lambda = 1064$ nm, $w_0 = 1$ mm, $z_0 = 0$ mm at $z = 2$ m. The l and m indices represent the number of dark lines in vertical and horizontal directions, respectively. Note, the width of the intensity distribution increases with indices l and m, while the peak intensity decreases. On the other hand, the phase is similar for all higher-order HG modes except for the sudden phase jump of π which is introduced in between intensity peaks. Besides, the phase of a higher-order HG mode shifts compared to the fundamental mode due to the Gouy phase. The Gouy phase for higher-order modes relates to the fundamental mode Gouy phase according to $\Phi_{lm} = \left(l + m + 1\right) \Phi_{00}$. The rings which are visible in the phase images originate from the curved wavefront as described in equation 2.13. The phase at a large radial distance of the wavefront lags the phase of the center, thus introducing a phase difference depending on the radial coordinate. A further discussion on this phase difference can be found in section 4.2.1.

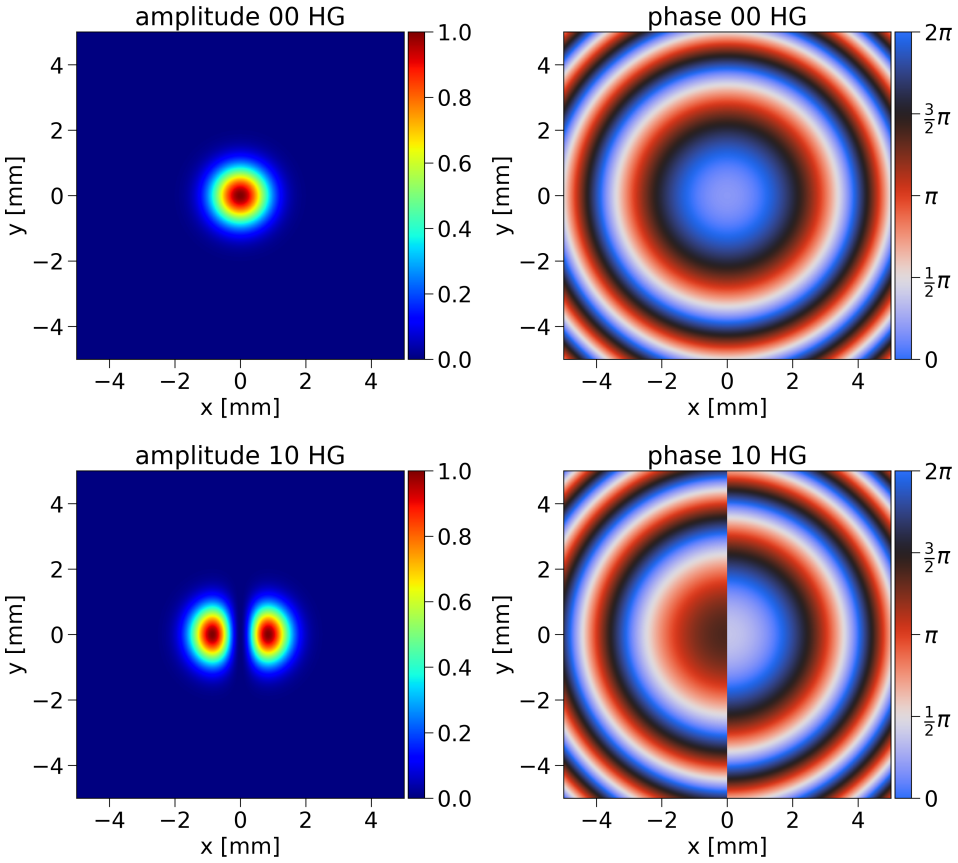


Figure 2.4: Amplitude and phase images for the fundamental mode $U_{00}(x,y,z)$ (upper row) and first higher Hermite-Gaussian mode $U_{10}(x,y,z)$ (bottom row). The modes are for a beam with $\lambda = 1064$ nm, $w_0 = 1$ mm, $z_0 = 0$ mm at $z = 2$ m.

2.1.5. Transformation and magnification by simple lenses

Gaussian beams propagating in free space are described in the previous sections. However, several elements are present in an optical system. For example, lenses can transform and magnify beams of light. In the thin lens approximation, the effect of a lens can be described with the lens-maker's formula for uniform spherical waves

$$\frac{1}{s} + \frac{1}{s''} = \frac{1}{f}, \tag{2.23}$$

where s is the object distance, s'' the image distance and f the focal length of the lens. To use this type of equation for a Gaussian beam the waist of the input beam has to be regarded as the object and the waist of the output beam as the image. Besides, the beam radius and radius of curvature vary according to equations 2.12 and 2.13. Therefore, the lens-maker's formula has to be altered for spherical Gaussian beams into (Self, 1983)

$$\frac{1}{s + \frac{z_R^2}{s-f}} + \frac{1}{s''} = \frac{1}{f}. \tag{2.24}$$

Figure 2.5 shows the normalized image distance as a function of the normalized object distance for different ratios of z_R/f. For $z_R/f \to 0$, this formula reduces to the standard lens-maker's formula with an asymptote at $s/f = 1$, meaning that the image distance becomes infinite for an object in the focal point, which results in a collimated beam. For $z_R/f > 0$ there is no asymptote and all curves pass through $s/f = s''/f = 1$. Therefore, there will always be a (virtual) waist on both sides of the lens. The lenses used in the phase camera set-up have focal lengths in the order of tens of centimeters while

the beam sizes are of the order of one millimeter. This results in z_R/f ratios of 5 and larger. Hence the image distances will be finite and of the order of the focal lengths.

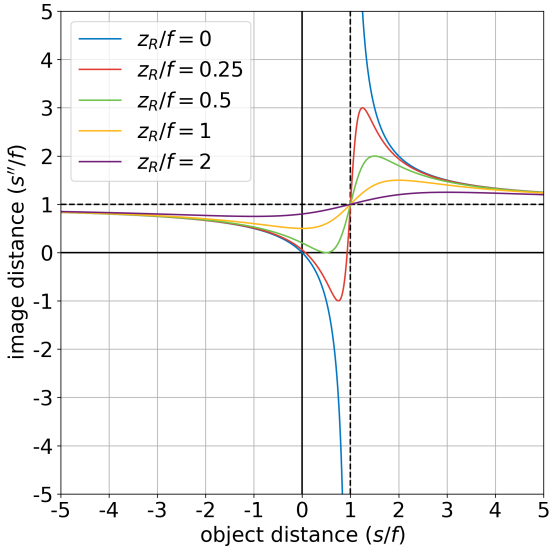


Figure 2.5: The normalized image distance (s''/f) as a function of the normalized object distance (s/f) for different normalized Rayleigh ranges (z_R/f).

A more general lens-maker's formula for Gaussian beams with the beam quality factor for real Gaussian beams M^2 can be written as (CVI Melles Griot, n.d.)

$$\frac{1}{s + \left(\frac{z_R}{M^2}\right)^2 \frac{1}{s-f}} + \frac{1}{s''} = \frac{1}{f}. \tag{2.25}$$

To calculate the effect of a series of optical elements, the initial Rayleigh range z_R, waist size w_0 and waist position z_0 need to be known. With the focal length of all elements, these parameters can be calculated throughout the system by using the magnification (Self, 1983)

$$m = \frac{w_0''}{w_0} = \frac{1}{\sqrt{\left(1 - \frac{s}{f}\right)^2 + \left(\frac{z_R}{f}\right)^2}}. \tag{2.26}$$

The Rayleigh range of the output beam of a lens is given by

$$z_R'' = m^2 z_R. \tag{2.27}$$

Figure 2.6 shows the magnification versus the normalized object distance for different ratios of z_R/f. Note the asymptote for $z_R/f = 0$ at $s/f = 1$, which represents the magnification for spherical waves. For $z_R/f > 0$ there is no asymptote and the magnification decreases with increasing z_R/f ratios. However, the magnification is still symmetric around $s/f = 1$.

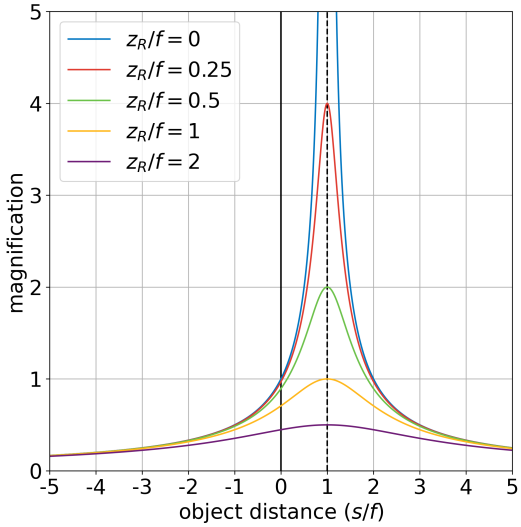


Figure 2.6: The magnification m as a function of the normalized object distance (s/f) for a different normalized Rayleigh ranges (z_R/f).

2.2. Sidebands

In the previous sections, a laser beam with a single optical frequency is assumed. However, in the phase camera set-up multiple frequencies are present. Sidebands are created in the laser field by an electro-optic phase modulator (EOM). In the Virgo gravitational wave detector, sidebands are used to control the distance between the mirrors and the angles of the mirrors with respect to the optical axis (Van der Schaaf, 2020). In this section, the creation of sidebands is mathematically derived (Oppenheim et al., 2013).

First, imagine the electric field $E(t)$ of a laser beam with amplitude A and frequency ω

$$E(t) = Ae^{i\omega t}. \tag{2.28}$$

An EOM is a crystal with electrodes on the top and bottom of the crystal. The electrical equivalent is a parallel plate capacitor. Across this crystal, a potential is applied that creates an electric field. This field changes the index of refraction of the crystal and thereby causes a phase shift in the laser beam (Saleh & Teich, 2009). For more information on EOMs see section 3.2.2. If a sinusoidally varying potential is applied with a small amplitude β and frequency Ω, then the phase of the laser beam becomes time-dependent

$$E(t) = Ae^{i\omega t + i\beta \sin(\Omega t)}. \tag{2.29}$$

The magnitude of the time-dependent phase β is small. Therefore, this expression can be simplified by a Taylor expansion

$$E(t) \approx Ae^{i\omega t}\left(1 + i\beta \sin(\Omega t)\right). \tag{2.30}$$

Now rewriting the sine into its complex exponential form gives

$$\begin{aligned} E(t) &\approx Ae^{i\omega t}\left(1 + \frac{\beta}{2}\left(e^{i\Omega t} - e^{-i\Omega t}\right)\right) \\ &= A\left(e^{i\omega t} + \frac{\beta}{2}e^{i(\omega+\Omega)t} - \frac{\beta}{2}e^{i(\omega-\Omega)t}\right). \end{aligned} \tag{2.31}$$

As a result, there are now three signals with frequencies ω, $\omega + \Omega$ and $\omega - \Omega$. This is interpreted as the initial carrier signal at ω and two newly created sidebands at $\omega + \Omega$ and $\omega - \Omega$. The amplitude β is a measure of the amount of modulation of the signal. Therefore, it is called the modulation index. If $\beta = 0$, then only the carrier signal is present. When $\beta = 1$, it is called 100% modulation.

In equation 2.30 the first-order Taylor expansion has been used. Therefore, only the first two sidebands are derived. If the Jacobi-Anger expansion is used instead, then the exponential can be written in terms of Bessel functions $J_k(\beta)$. As a result, it is found that there are infinitely many sidebands created at frequencies $\omega + k\Omega$ and $\omega - k\Omega$ with $k = 1, 2, .., \infty$

$$\begin{aligned} E(t) &= Ae^{i\omega t + i\beta \sin(\Omega t)} \\ &= Ae^{i\omega t}\left(J_0(\beta) + \sum_{k=1}^{\infty} J_k(\beta)\, e^{ik\Omega t} + \sum_{k=1}^{\infty} (-1)^k J_k(\beta)\, e^{-ik\Omega t}\right). \end{aligned} \tag{2.32}$$

Note that the sidebands are always created symmetrically around the carrier signal. However, their amplitudes become progressively lower with increasing k. For the used modulation index of this research $\beta \approx 0.12$, only the first upper and lower sideband are significant.

2.3. Beam profiling

To describe a propagating Gaussian beam in an optical set-up, the beam parameters need to be determined. The beam waist w_0, quality factor M^2 and waist position z_0 can be determined by measuring the beam profile at multiple positions along the beam with a beam profiler. In this research, the used beam profiler is a camera with a high resolution to measure the intensity distribution of a beam cross-section. For a theoretical Gaussian beam, this intensity distribution is described by equation 2.11. Figure 2.7 shows a screenshot of a measurement with a beam profiler (CinCam CMOS-1.001-Nano). Note that an ideal beam cross-section would be circular, but in this case, it is somewhat elliptic. This originates from astigmatism in the beam which is introduced due to surface defects of the components or misalignments in the optical set-up. The optical system is then not symmetric anymore around the optical axis. Therefore, two perpendicular planes have different focal points creating elliptic-shaped beam profiles.

In the beam profiling software, RayCi 64bit V2.3.4 (CINOGY Technologies, 2013), the radius is extracted from a fit to the beam profile. The radius is determined where the beam intensity has fallen to $1/e^2$ of its maximum. Since the beam spot is elliptic, the minor and major radii are not equal. These are indicated by red lines in figure 2.7.

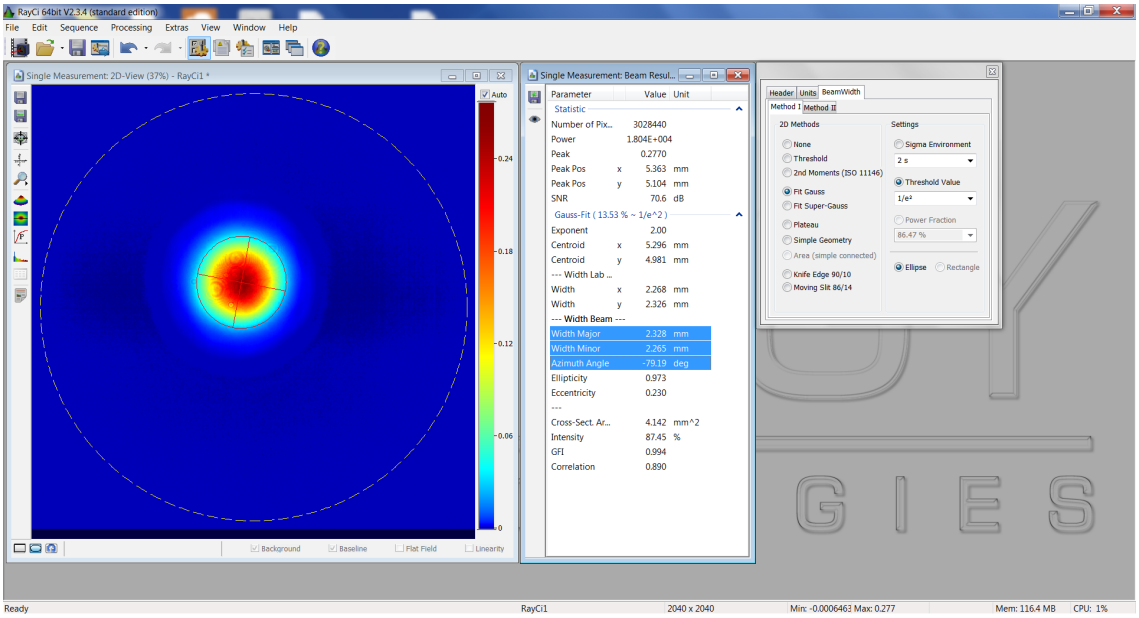


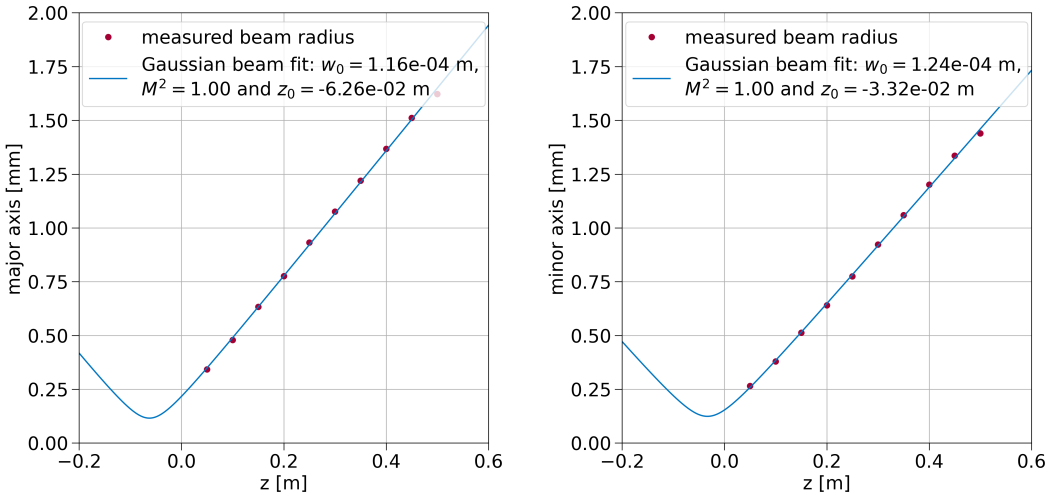
Figure 2.7: Screenshot of a measurement with the beam profiler software, RayCi 64bit V2.3.4 (CINOGY Technologies, 2013).

This measurement is repeated for multiple positions of the beam profiler along the axis of propagation. The next step is to plot the beam minor and major radii separately versus the positions of the beam

profiler as shown in figure 2.8. In these plots, the evolution of the beam radius can be seen. This behavior should be described by equation 2.16 for a Gaussian beam with its waist at $z_0 = 0$. Since the z-position of the beam waist has an arbitrary offset with respect to the lab coordinate system, this equation is altered into

$$w_R(z) = w_{0R}\sqrt{1 + \left(\frac{\lambda (z - z_0) M^2}{\pi w_{0R}^2}\right)^2}. \tag{2.33}$$

This relation is fitted with a least squared optimization to the measured major and minor beam radii. The fitted function to the data points is also shown in figure 2.8 as a solid line. The resulting beam parameters for this laser beam are shown in the legends in the figures. Note that the quality factor for this fit is approximately equal to a theoretical Gaussian beam.



(a) major axis (b) minor axis

Figure 2.8: Measured beam radius along the major (a) and minor (b) axis at different positions along the direction of propagation. The blue solid line is fitted to determine the Gaussian beam parameters which are shown in the inside legends.

In the figures above, the minor and major radii have slightly different beam waists and waist positions. This is due to the earlier described astigmatism.

With the beam profiling software, the major and minor axis of the beam spot are measured instead of the size in the x- and y-direction, because the orientation of the astigmatism with respect to the coordinate system is unknown. However, this is only possible if the orientation of the major axis with respect to the x- and y-direction does not change as a function of the profiler position. Therefore, the azimuth angle between the major axis and the y-axis is measured too. A jump of 90 degrees occurs between the waists of both directions, at the point where the beam sizes are the same. Then the major and minor axis are swapped. In the example measurement of the previous figures, the azimuth angle is constant. Hence, the measured major and minor axis data is not mixed.

3

Phase camera

A phase camera is a diagnostic tool that was invented for studying gravitational wave detectors (Goda et al., 2004). With this device, the amplitude and phase of a laser wavefront can be imaged. Because heterodyne demodulation is applied different modulation frequencies can be studied at the same time. In this chapter, the working principle of the phase camera is discussed as well as the optical set-up used in this research to study its performance. Subsequently, the different components of the phase camera set-up are discussed more in-depth. In the end, the beam conditioning on the final beam splitter is described.

3.1. Working principle of the phase camera

The basic principle of the studied phase camera in this thesis is to scan a laser beam over a small photodiode and then demodulate the signal from the different measurement points sequentially. As shown in figure 3.1, sidebands are created with an electro-optic modulator (EOM) which phase modulates the laser beam. The sidebands that appear because of the modulation are used in the interferometer to control the position and alignment of the mirrors in the interferometer (Van der Schaaf, 2020). A small fraction of the light circulating in the interferometer is picked off and sent to the phase camera as a test beam. The test beam is recombined after the scanner with a reference beam. This reference beam is split from the initial beam before the EOM and then shifted in frequency by 80 MHz by an acousto-optic modulator (AOM). A beam splitter (BS) is used to recombine the test and reference beams. The interference of the test and reference beam yields (amplitude modulated) beat signals at the sum and difference of the heterodyne and sideband frequency. These intensity fluctuations are detected with a photodiode (PD). The photodiode signal is digitized with a fast, high-resolution ADC. Finally, the demodulation is done digitally by an FPGA box and further data processing is done in a system similar to the Advanced Virgo real-time system (Agatsuma et al., 2019).

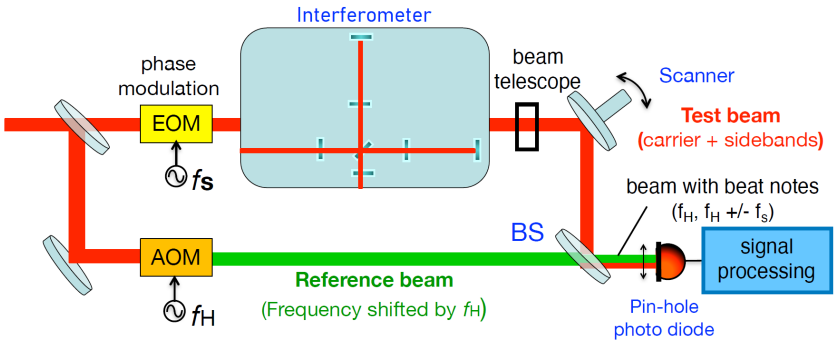


Figure 3.1: Working principle of the phase camera. Adapted from Van Beuzekom et al. (2019).

In this thesis, the scanner scans only the test beam while the reference beam is fixed, this is called one-beam scanning. There is another configuration which is called two-beam scanning. In this configuration, the reference and test beam are recombined before the scanner and are both scanned simultaneously. For one-beam scanning the power of the reference beam is constant which is an advantage over two-beam scanning since this results in a higher signal-to-noise ratio. On the other hand in one-beam scanning, there is an additional phase shift of the wavefront caused by path length differences in the path between the scanner and PD, which is a disadvantage compared to two-beam scanning (Agatsuma et al., 2019).

3.2. Optical set-up

For this research, a prototype set-up of the Virgo phase camera is built in the optical lab at Nikhef. In figure 3.2 the optical layout of the set-up is shown. The used laser is a 1064 nm Mephisto 500 mW YAG Laser System by Coherent. This laser is shared by different set-ups on the optical table, and the laser power for the phase camera set-up is approximately 0.140 W. At the entrance of the set-up an iris is placed to remove potential beam halo due to reflections. Next, beam splitter BS1 (Thorlabs BSW11) splits the beam to create the reference beam which is focused by lens L1 with $f = 500$ mm (Thorlabs LA1908-C). After which two mirrors M1 and M2 (IDEX Optical Technologies Y1-1025-45) steer the beam before it enters the single mode optical fiber (Thorlabs P3-1064PM-FC-1) via a Triplet Collimator (Thorlabs TC18APC-1064). Through this fiber, the beam is led to the acousto-optic modulator (AA Opto-Electronic MT80-IR60-Fio-PM0,5-J3V-A).

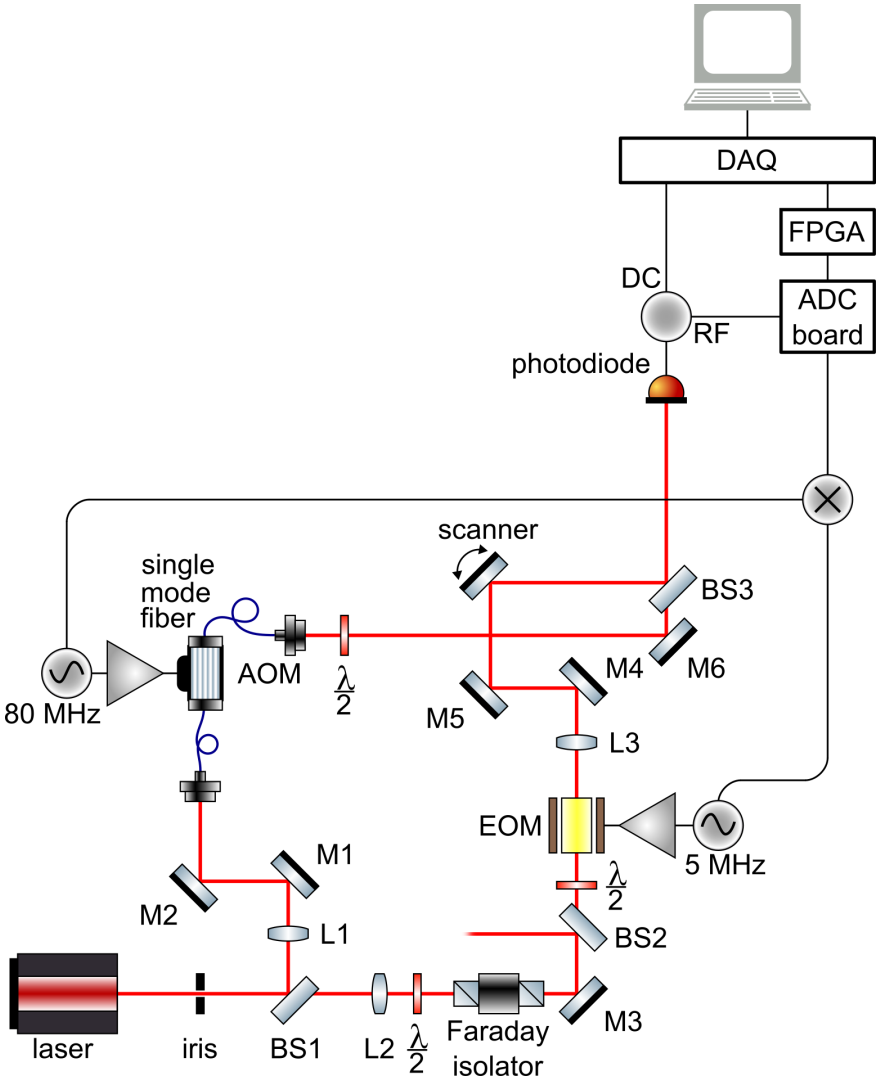


Figure 3.2: Schematic representation of the optical set-up of the prototype phase camera at Nikhef. Created with Franzen (2012).

The test beam which is split from the reference beam at the initial beam splitter BS1 is focused with a $f = 200$ mm lens L2 (Thorlabs LB1945-C). After which the polarization is rotated by a half-wave plate (Thorlabs WPH10M-1064) to align it with the entrance polarization for the Faraday isolator (Thorlabs IO-5-1064-HP). This optical diode is used to protect the laser from unwanted reflected light. Next, mirror M3 (Thorlabs BB1-E03) is used to steer the beam to a second beam splitter BS2 (Thorlabs BSW11) which can be used to create a beam that bypasses the electro-optic modulator such that aberrations can be introduced. The test beam continues to another half-wave plate (Thorlabs WPH10M-1064) to rotate the polarization before entering the electro-optic modulator (Qubig PS2M-NIR). After which a $f = 300$ mm lens L3 (Thorlabs LB1779-C) is placed to focus the beam (see section 3.3). Next, two mirrors M4 and M5 (Thorlabs BB1-E03) are used to lead the beam to the scanner (Physik Instrumente S-334-2SL). This tip/tilt mirror moves to scan the whole test beam. Finally, the beam reaches the beam splitter plate BS3 (Thorlabs BSW11) which recombines the beam with the reference beam coming from the acousto-optic modulator via a single mode optical fiber (Thorlabs P3-1064PM-FC-1), an output collimator (Thorlabs TC12APC-1064) and mirror M6 (Thorlabs BB1-E03). The polarization of the reference beam is rotated by a half-wave plate (Thorlabs WPH10M-1064) to align it with the polarization of the test beam. The recombined beam reaches the photodiode (OSI Optoelectronics FCI-InGaAs-55) at the end which measures the beat signal created.

The measured signal of the photodiode is amplified and split into a DC and a RF component. The RF component is digitized by an ADC board (Intersil ISLA214P50) and the DC signal is acquired by a slow ADC unit of the Virgo real-time system. The signals from the waveform generators that drive the AOM and EOM are split and a copy of them is also digitized by the same ADC board. These digital signals are then demodulated with a FPGA, field-programmable gate array (Xilinx Virtex-7 XC7VX485). This is a heterodyne demodulation that results in intensity and phase information for maximally 11 frequencies. Finally, this data is combined in the data acquisition box (DAQ) with the measured positions of the scanner to create the images.

3.2.1. Acousto-optic modulator

An acousto-optic modulator (AOM) is a crystal which is vibrating due to sound waves (McCarron, 2007). These waves are produced by a piezoelectric transducer which is attached to the crystal and vibrates due to an oscillating electric signal. The incoming light scatters off the vibrating crystal and a diffraction pattern emerges. The constructive interference occurs at scattering angles $m\theta$ with m an integer and θ the first scattering angle with constructive interference. This momentum change of the light is caused by energy exchange with the phonons in the crystal. Since for light, energy is directly related to frequency, this causes frequency shifts of $mf_{crystal}$ for the scattered light with $f_{crystal}$ the frequency at which the crystal vibrates due to the generator signal. Note that the chosen frequency is close to the resonance frequency of the crystal to make the conversion efficient. To shift the frequency of a laser beam with $f_{crystal}$, the output of an AOM has to be placed exactly under an angle θ with respect to the incoming beam. The amount of light diffracted by the crystal depends on the number of phonons in the crystal. Hence, the conversion efficiency of the AOM (P_{out}/P_{in}) depends on the amount of RF power. Note that there is power loss in an AOM because the initial beam power is spread out over the diffraction pattern.

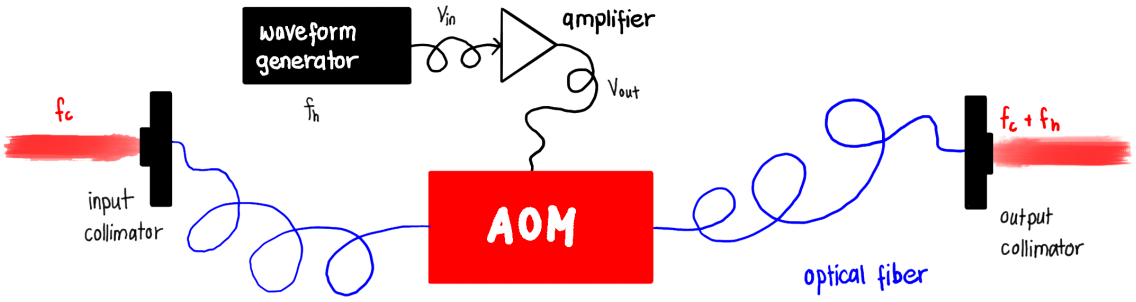


Figure 3.3: Overview of the set-up around the AOM. Adapted from Van der Schaaf (2020).

In the prototype set-up an AA Opto-Electronic MT80-IR60-Fio-PM0,5-J3V-A is used. An input collimator is used to focus the input beam ($P_{in} = 82$ mW) into a single-mode optical fiber which leads the beam to the AOM. The AOM is driven by a waveform generator (Keysight 336090A Series) which creates an 80 MHz sinusoidal signal with an amplitude of 6 dBm. This signal is split (Mini-Circuits ZFSC-2-4+), where half of it goes to the AOM and the other half to the ADC and FPGA for demodulation. The part going to the AOM is amplified by about 24 dB (Mini-Circuits ZHL-3A). This amplifier is powered by a 24 V power supply (Aim-TTi EL302). The output fiber of the AOM leads the frequency-shifted beam to the output collimator. The output power is $P_{out} = 16.8$ mW.

3.2.2. Electro-optic modulator

In an electro-optic modulator (EOM), an electric field is applied to a material causing the refractive index to change due to an electro-optic effect. This electric field is created by plates across the material in the EOM, the electrical equivalent of this structure is a parallel plate capacitor. The change in refractive index causes an optical phase shift. If a sinusoidal signal is applied to the plates, the electric field and thus the refractive index will vary periodically too (Saleh & Teich, 2009). This causes a time-dependent phase shift which results in the creation of sidebands as shown in section 2.2.

The EOM in the prototype set-up at Nikhef is the Qubig PS2M-NIR. It is powered by a sinusoidal signal of 5 MHz created by the same waveform generator as the AOM (Keysight 336090A Series) but on the second output port. The output of 3 dBm is split into two paths (Mini-Circuits ZFSC-2-4+), one to drive the EOM and the other as an electrical reference signal to the ADC and FPGA for demodulation. The EOM branch is first amplified by about 24 dB (Mini-Circuits ZHL-3A+). This amplifier is powered by a 24 V power supply (Aim-TTi EL302).

Before the EOM a half-wave plate (Thorlabs WPH10M-1064) is placed since the EOM is polarization dependent. The input beam polarization needs to be aligned with one of the optical axes of the crystal in the EOM. After this half-wave plate, the beam enters the EOM via a 2 by 2 mm^2 aperture. To optimize the output power of the EOM the half-wave plate is mounted on a rotation stage (Thorlabs RSP1X15/M). After the EOM a power meter (Thorlabs S130C) is used to measure the output power of the EOM. The output power of the EOM equals 19.9 mW for a 20.0 mW input beam.

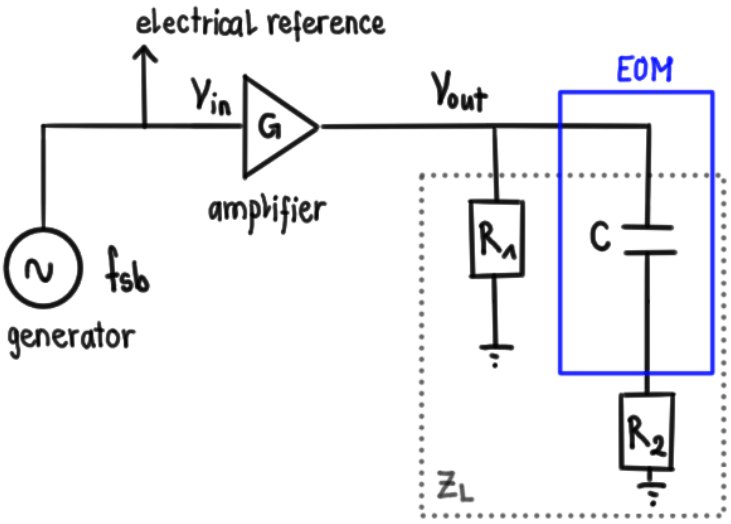


Figure 3.4: Sketch of the electric circuit which drives the EOM. Adapted from Van der Schaaf (2020).

The sinusoidal signal is applied to one of the connectors of the EOM. The other connector is terminated with a 50 Ω resistor (R_2) see figure 3.4. To match the load impedance with the output impedance of the amplifier (50 Ω), a resistor ($R_1 = 50$ Ω) is placed parallel to the EOM. The impedance of the load Z_L,

EOM and resistors, can be calculated as

$$\begin{aligned}\frac{1}{Z_L} &= \frac{1}{R_1} + \frac{1}{\frac{1}{j\omega C} + R_2} \\ &= \frac{1}{R_1} + \frac{j\omega C}{1 + j\omega R_2 C} \\ &= \frac{1 + j\omega\left(R_1 + R_2\right) C}{R_1\left(1 + j\omega R_2 C\right)},\end{aligned} \tag{3.1}$$

with j the imaginary unit, ω =2π·5 MHz the angular frequency and $C = 14$ pF the EOM capacitance. Now inverting the fraction gives a load impedance of

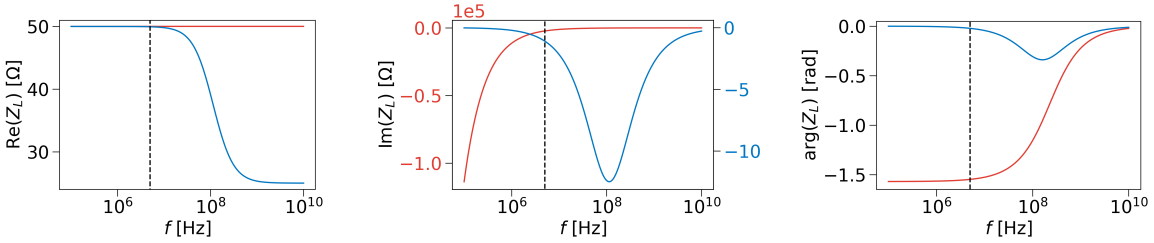
$$\begin{aligned}Z_L &= \frac{R_1\left(1 + j\omega R_2 C\right)}{1 + j\omega\left(R_1 + R_2\right) C} \\ &= \frac{R_1\left(1 + j\omega R_2 C\right)}{1 + j\omega\left(R_1 + R_2\right) C} \frac{1 - j\omega\left(R_1 + R_2\right) C}{1 - j\omega\left(R_1 + R_2\right) C} \\ &= \frac{R_1\left(1 - j\omega\left(R_1 + R_2\right) C + j\omega R_2 C + \omega^2 R_2\left(R_1 + R_2\right) C^2\right)}{1 + \omega^2\left(R_1 + R_2\right)^2 C^2} \\ &= \frac{R_1 + \omega^2 R_1 R_2\left(R_1 + R_2\right) C^2}{1 + \omega^2\left(R_1 + R_2\right)^2 C^2} - j\frac{\omega R_1^2 C}{1 + \omega^2\left(R_1 + R_2\right)^2 C^2}.\end{aligned} \tag{3.2}$$

This results in a load impedance of $Z_L \approx 49.9 - 1.1j\ \Omega$. Note that without the resistor R_1 the load impedance would be

$$Z_L = \frac{1}{j\omega C} + R_2, \tag{3.3}$$

which gives $Z_L \approx 50 - 2300j\ \Omega$. Therefore, the load impedances match best with resistor R_1 parallel to the EOM.

The load impedance is frequency dependent. To validate if the input resistor R_1 is needed for different frequencies, equation 3.2 is plotted depending on the frequency. The real and imaginary parts as well as the argument of the load impedance are plotted separately in figure 3.5. It can be seen that for frequencies up to approximately 10 MHz, it is necessary to use a parallel input resistor. For higher frequencies, the real and imaginary parts are of the same order of magnitude resulting in an impedance mismatch. In this case, this configuration is not suitable. Note that above 1 GHz, the impedance matches again, but the EOM is limited to a frequency of 150 MHz (Qubig, 2020).



(a) Re(Z_L) (b) Im(Z_L) (c) arg(Z_L)

Figure 3.5: Frequency dependence of the real part (a), imaginary part (b) and argument (c) of the load impedance of the EOM and resistors. The blue and red lines are the load impedance with and without input resistor R_1, respectively. The black dashed line is the operational EOM frequency of 5 MHz.

3.2.3. Scanner

To scan the wavefront of the test beam over the photodiode a tip/tilt mirror is used, which is placed on piezo elements (Physik Instrumente S-334-2SL). Figure 3.6 shows the set-up around the scanner as well as a picture of the real scanner. Two voltages, $G \cdot V_1$ and $G \cdot V_2$, can be applied to the piezo elements to tilt the scanner mirror in different directions. The actuation voltage of the piezo elements lies between 0 and 100 V. The mirror is tilted in one direction for voltages below 50 V and in the opposite direction for voltages above 50 V (Physik Instrumente, 2003). The Virgo DACs (Analog Devices AD1955) create the input voltages V_1 and V_2 which are amplified with a gain $G \approx 10.0$ (Physik Instrumente E-505.00). These amplifiers take an input between -2 and +12 V and a DC offset between 0 and 10 V. The output range of these amplifiers is from -30 to +130 V (Physik Instrumente, 2013). To tilt the mirror in both directions the DC offsets should be 5 V, such that positive and negative DAC voltages tilt the mirror in opposite directions. To measure the mirror position, two strain gauges S_1 and S_2 are used. The controllers of these sensors (Physik Instrumente E-509.X3) have an output between 0 and 10 V (Physik Instrumente, 2009), which are read out via Virgo ADCs (Analog Devices LTC2378-20).

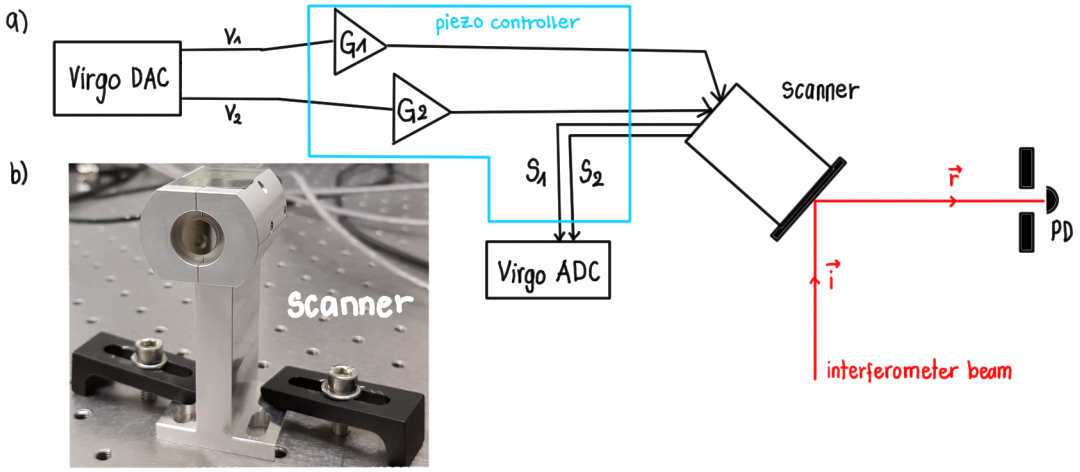


Figure 3.6: Schematic representation of the set-up around the scanner (a) as well as a picture of the scanner in the real set-up (b). Adapted from Van der Schaaf (2020).

To convert the strain gauge measurements into angles the scanner needs to be calibrated. First, the scanner angles are measured for DC input voltages. To measure the scanner angles, a beam profiler (CinCam CMOS-1.001-Nano) is set at a distance of $L = 0.75$ m. The beam profiler provides the coordinates of the center of the reflected beam for varying DC input voltages. Figures 3.7a and 3.7b show the coordinates of the center of the beam when the scanner is actuated on channel 1 and 2, respectively. Note that the scanner actuation signals (S_1,S_2) are not aligned with the x and y axes which are both orthogonal to the direction of propagation of the reflected beam. Therefore, both the x and y coordinate of the beam spot position alter when an input voltage is applied to only one of the inputs of the scanner. Only the slope is relevant for these plots since the offset depends on the alignment of the beam profiler. Therefore, a linear function is fitted with a least squared optimization to find the dependency of x and y on V_1 and V_2:

$$\begin{aligned} \Delta x &= 1.81 \cdot V_1 \quad \text{and} \quad \Delta y = 1.36 \cdot V_1, \qquad \text{[mm/V]} \\ \Delta x &= 1.65 \cdot V_2 \quad \text{and} \quad \Delta y = -1.28 \cdot V_2. \qquad \text{[mm/V]} \end{aligned} \tag{3.4}$$

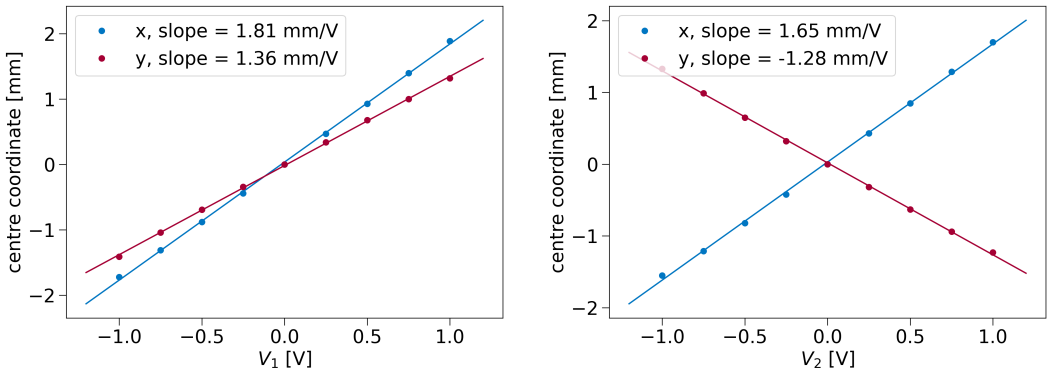
Next coefficients α_1 and α_2 can be defined, which relate the tilt angles to the input voltages as

$$\Theta_1 = \alpha_1 \cdot V_1 \quad \text{and} \quad \Theta_2 = \alpha_2 \cdot V_2. \tag{3.5}$$

Since the scanner axes are not aligned with the x,y-axis system, the pitch Θ_p and yaw angles Θ_y of the mirror are not the same as Θ_1 and Θ_2. According to Van der Schaaf, 2020, the conversion between the two axis systems equals

$$\Theta_y = \sqrt{2}\,(\Theta_2 - \Theta_1)\,, \tag{3.6}$$

$$\Theta_p = \Theta_2 + \Theta_1. \tag{3.7}$$



(a) channel 1 (b) channel 2

Figure 3.7: Calibration of the tilt of the scanner when applying an input voltage. The coordinates of the center of the beam measured with a beam profiler are plotted versus the input voltages for channel 1 (a) and channel 2 (b) of the scanner.

The yaw and pitch angles are maximal ± 25 mrad (Physik Instrumente, 2003). Therefore, these angles can be approximated by $\Theta_y \approx \frac{x}{L}$ and $\Theta_p \approx \frac{y}{L}$. Resulting in

$$\alpha_1 = \frac{1}{2\sqrt{2} \cdot V_1} \left(\sqrt{2} \cdot \Theta_p - \Theta_y\right) \approx \frac{1}{2\sqrt{2} \cdot V_1 L} \left(\sqrt{2} \cdot y - x\right), \tag{3.8}$$

$$\alpha_2 = \frac{1}{2\sqrt{2} \cdot V_2} \left(\sqrt{2} \cdot \Theta_p + \Theta_y\right) \approx \frac{1}{2\sqrt{2} \cdot V_2 L} \left(\sqrt{2} \cdot y + x\right). \tag{3.9}$$

Here L is the distance from the scanner mirror to the photodiode and $L = 850$ mm. Now combining these two equations for α_1 and α_2 with the fitted slopes in equation 3.4 gives $\alpha_1 \approx 1.76$ mrad/V and $\alpha_2 \approx -1.63$ mrad/V. The obtained values are a factor three lower than the expected 5 mrad/V (Physik Instrumente, 2003).

Now that the relation between the angle and control voltage is known, the strain gauges can be calibrated. Again a DC voltage is applied to the scanner, but now the strain gauges are measured with a Virgo ADC. The measurements are fitted to a linear function with a least squared optimization to get the relation between the scanner actuation voltages (V_1, V_2) and the strain gauge signals (S_1, S_2). Figure 3.8 shows the measurements together with the fitted functions. The strain gauges relate to the input voltages as

$$V_1 = 57.2 S_2 - 15.4 \text{ V} \quad \text{and} \quad V_2 = 61.6 S_1 - 15.2 \text{ V}. \tag{3.10}$$

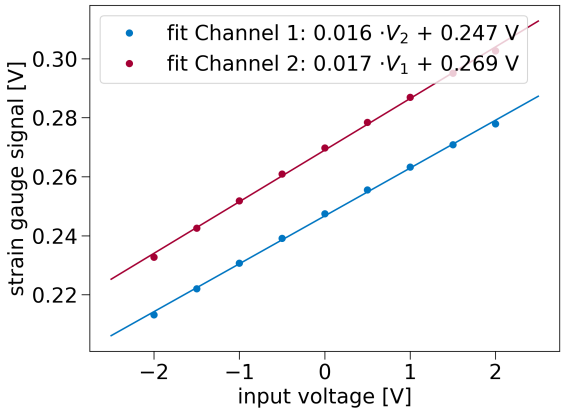


Figure 3.8: Strain gauge dependence on the input voltage of the scanner.

If the two calibrations are combined, the tilt of the scanner can be calculated from the measured strain gauge signals. This can be used to reconstruct an image from a group of measurement points. The scanner pattern describes an Archimedean spiral to scan the wavefront of the test beam:

$$x(t) = R_{img} \cdot \frac{t}{T_{img}} \cdot \cos(2\pi \cdot f_{scan} \cdot t) \quad \text{and} \quad y(t) = R_{img} \cdot \frac{t}{T_{img}} \cdot \sin(2\pi \cdot f_{scan} \cdot t), \tag{3.11}$$

with R_{img} the radius of the image (typically a few millimeters), T_{img} the acquisition time for one image and f_{scan} Hz the spiral revolution frequency. This scanning pattern is used since it gives a smooth movement for the mirror. It prevents sudden changes, which could result in the mirror overshooting its intended position.

Along this trajectory measurements are done. The sampling time of the ADC is 2 ns and there are taken 16384 ADC samples per pixel. In total, there are $N_{img} = 16384$ pixels measured in one full spiral. So one measurement takes $T_{img} \approx 0.537$ s. After this time the mirror moves back inward again in 0.01 s and stays in the center position until the next measurement starts at the turn of the next second. The exact time is obtained from the so-called IRIG-B signal from GPS. The total scan time is less than 1 s because all data in the Virgo DAQ system is stored in frames with a duration of 1 second (Agatsuma et al., 2019). Note that the scanning frequency influences the distribution of the pixels in the spiral. For this configuration, the optical scanning frequency is $f_{scan} = 120$ Hz (Van der Schaaf, 2020). Along the diameter of the circular image, there are $n \approx \sqrt{N_{img}}$ pixels. Therefore, there are $n_{cycle} = \sqrt{N_{img}}/2$ cycles within one scan. This corresponds with $N_{img}/n_{cycle} \approx 2 \cdot n$ pixels per cycle. The pixels are equally spaced angularly, thus the radial distance between two pixels at the edge is $\pi R_{img}/n \approx 60$ μm. Since this distance is of the order of the photodiode size (55 μm), the whole wavefront is imaged.

3.2.4. Photodiode

The beat signal is measured by a photodiode (OSI Optoelectronics FCI-InGaAs-55). This pin-hole detector has an active diameter of 55 μm (OSI Optoelectronics, n.d.) and is housed in a brass box for electromagnetic shielding reasons. In the photodiode box, the DC and RF signals are split and amplified. The RF signals are transported to the ADC board where they are digitized with a 14-bits 500 MS/s ADC and then digitally demodulated at up to 11 frequencies in parallel. The DC signals are acquired by Virgo-style ADCs and are primarily used for alignment purposes.

3.2.5. Demodulation

The demodulation of the beat signal takes place in a field-programmable gate array (FPGA). This box takes the RF signal of the photodiode box as input as well as a copy of the signals which drive the EOM and AOM. These signals are all analog and are first digitized by a 14-bit ADC which is placed before the FPGA. Digital demodulation is used instead of analog demodulation since this allows demodulation of many sidebands in parallel (Agatsuma et al., 2019). After digitization, each point pixel is a time series with 16384 samples. A Hann window is applied to the series after which they are multiplied by a sine and cosine at the demodulation frequencies. Next, the signal is time integrated by summing the data. These signals are the in-phase (I) and quadrature (Q) components of the measured signal. Finally, the amplitude is found by taking the square root of the sum of the squared I and Q components. To get the phase the arctangent of Q over I is taken. The same demodulation process is used for the mixed AOM and EOM digitized signals which provide the phase reference. The reference phase is subtracted from the measured phase to get the relative phase of the measured wavefront (Van der Schaaf, 2020).

This FPGA demodulates the RF signal at maximal 11 frequencies. The demodulation frequencies can be changed to any frequency between 1 and 250 MHz. In this thesis, only the carrier frequency (80 MHz) and the first upper and lower sideband (75 and 85 MHz) are studied. The demodulated phase and amplitude points are transferred to the DAQ to be saved in frame files of 1 s. In these frame files also the strain gauges measurements are stored. Frames are grouped in batches of 100 frames and stored on disk to later create images. Finally, a Python code has been written to calculate the x, y-coordinates of each pixel from the strain gauge signals via the described calibration (see section 3.2.3). Next, the amplitude and phase points are mapped to a 100 x 100 grid to create the images of a phase camera. Since the scanner pattern is an Archimedean spiral the corners of the image contain no data and are set to zero. There are also image pixels with multiple measured points. In this case, the average is

taken of the amplitude measurements. The phase value of such a pixel is calculated by taking the angular average (Rudakova, 2018)

$$\phi_{avg} = \arctan\left(\frac{\sum_i^N \sin(\phi_i)}{\sum_i^N \cos(\phi_i)}\right), \tag{3.12}$$

where ϕ_{avg} is the phase average and ϕ_i the phase of measurement i in this single image pixel. The Python code which reads the batched frame files and creates these intensity and phase images is attached in appendix B.

An example of a measurement result of the phase camera prototype set-up can be seen in figure 3.9. The carrier frequency (middle) and the upper (right) and lower (left) sideband are shown. In the upper row, the intensity images are shown, while in the bottom row the phase images. The wavefront is scanned over a radius of approximately 2 mm. Note that for the phase images, the pixels with an intensity lower than a threshold (100 a.u.) are made black. This is to filter out pixels with a low signal-to-noise ratio. In the measurement, it can be seen that the intensity images all look similar only the intensity differs by a few orders of magnitude between the carrier and the sidebands. The peak in the intensity images is almost circular as expected from a Gaussian beam without aberrations. The rings seen in the phase images are mostly due to the scanning set-up which is explained in chapter 4. The phase image of the carrier frequency is blurred on the left edge because the measured power is lower in these pixels. Therefore, the noise in those pixels is larger. To prevent this, the intensity and phase images should both be centered. A further discussion on this centering can be found in chapter 6.

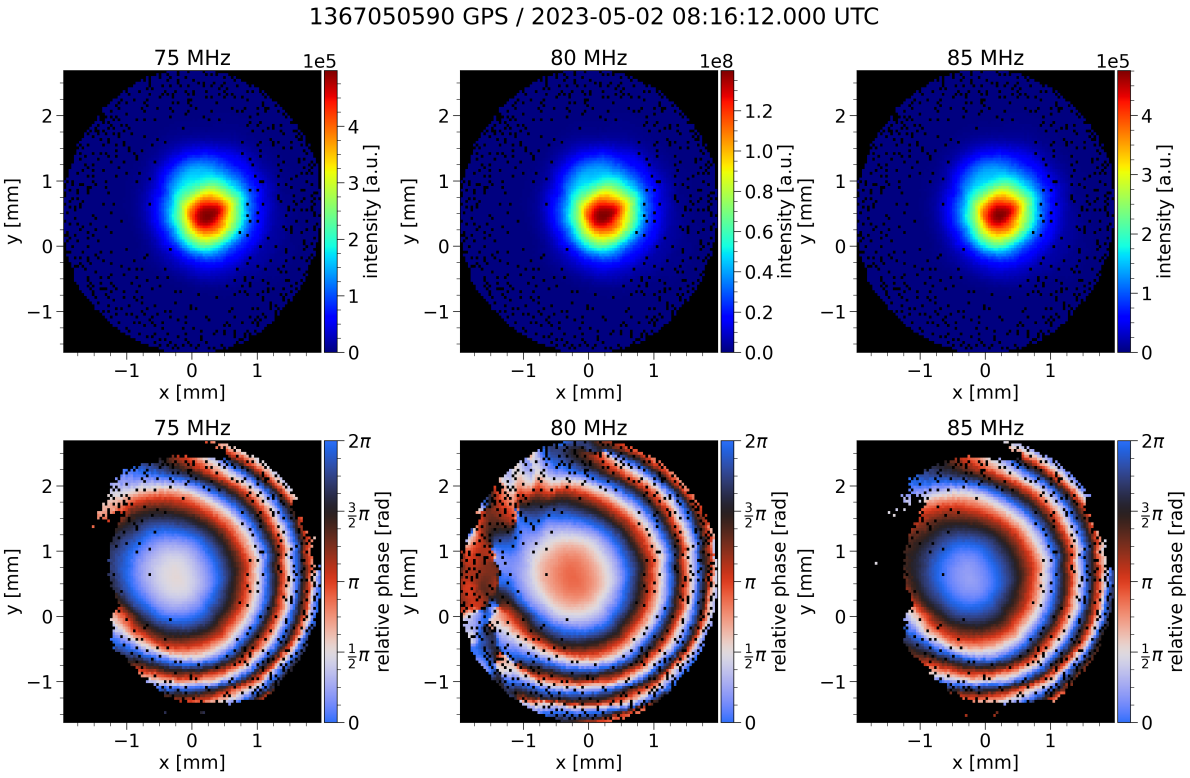


Figure 3.9: Example measurement of the phase camera. The top row shows from left to right the intensity images of the lower sideband, carrier and upper sideband. The bottom row shows the corresponding phase images at the three demodulated frequencies.

3.3. Beam conditioning

The beat signal which is measured with the phase camera is created at the final beam splitter. At this surface, the beams coming from the AOM and EOM recombine. Both beams are Gaussian beams with a differently curved wavefront. Therefore, the phase difference between both beams is not constant on the beam splitter surface. To remove this effect in the phase images, the wavefront should have the same radius of curvature. This is accomplished by conditioning the beams such that the waists are positioned at the beam splitter surface. The AOM beam is focused by the output collimator and is therefore already collimated with a waist size of approximately 1.1 mm. EOM beam on the other hand strongly diverges.

To condition the EOM beam such that the waist is located at the beam splitter, a lens is needed. The beam properties are determined by measuring the beam size at different positions along the beam with a beam profiler. For a detailed description of this measurement see section 2.3. The measured beam waists of the EOM beam are $w_{0,major} = 0.13$ mm and $w_{0,minor} = 0.12$ mm, the waist positions $z_{0,major} = -0.81$ m and $z_{0,minor} = -0.83$ m and the quality factors $M^2_{major} = 1.01$ and $M^2_{minor} = 1.01$. The software package JamMt (Thüring & Lastzka, 2011) is used, which is a mode-matching tool to simulate the behavior of Gaussian beams. Different configurations with different lenses are simulated for both the measured major and minor axis. Figure 3.10 shows a screenshot of JamMt with the final position for the lens and the laser beam along the major axis. The dashed green lines are the positions of the lens, mirrors, scanner, beam splitter and photodiode. The chosen lens has a focal length of 300 mm. The simulated beam sizes behind the lens along the major axis at different positions are shown in table 3.1. Note that the waist indeed is located near the beam splitter position.

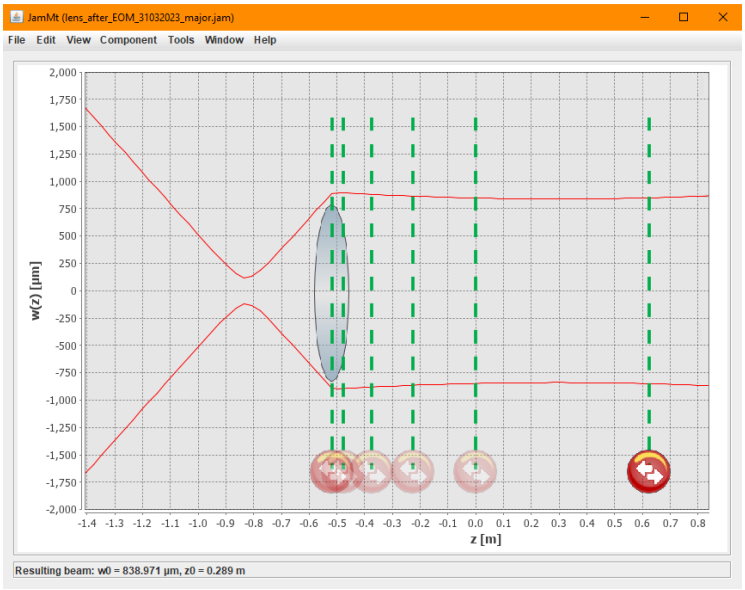
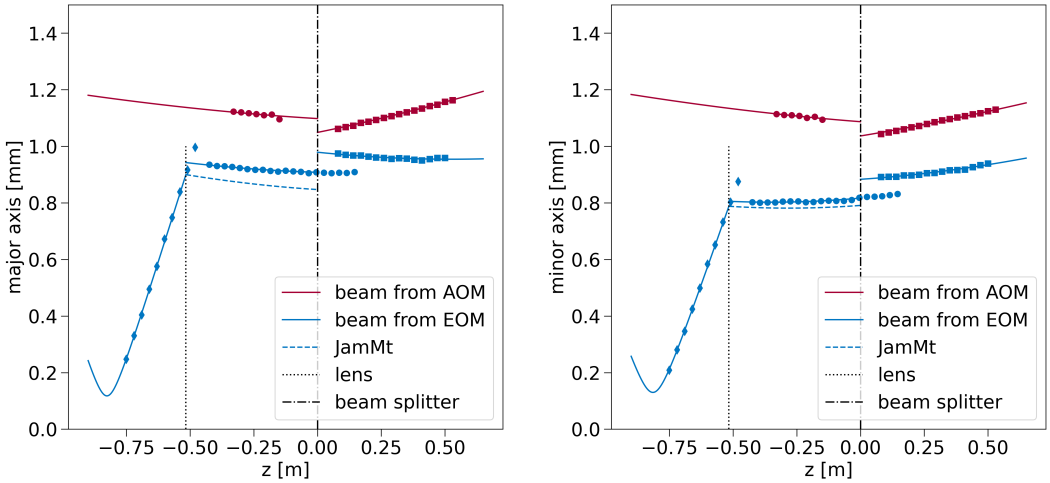


Figure 3.10: Screenshot of a simulation with JamMt (Thüring & Lastzka, 2011) to determine the position of a lens in a Gaussian beam.

Table 3.1: Simulated beam sizes along the major axis at different positions behind a $f = 300$ mm lens for the beam coming from the EOM.

	position [m]	beam size [μm]
lens	-0.516	899.728
mirror 1	-0.475	893.813
mirror 2	-0.375	880.706
scanner	-0.225	864.213
beam splitter	0	847.023
photodiode	0.625	849.889

After the JamMt simulations, the lens was placed in the set-up. To check whether the behavior of the beam is as simulated, the beam properties are measured after the lens too. In figure 3.11 the measured major and minor beam radii of the beam before and after the lens are plotted in blue. The simulated beam after the lens is also plotted as the dashed blue line. Note that the measured radii are slightly larger but the trend is very similar. This means that the lens is placed at approximately the position as determined by simulation.



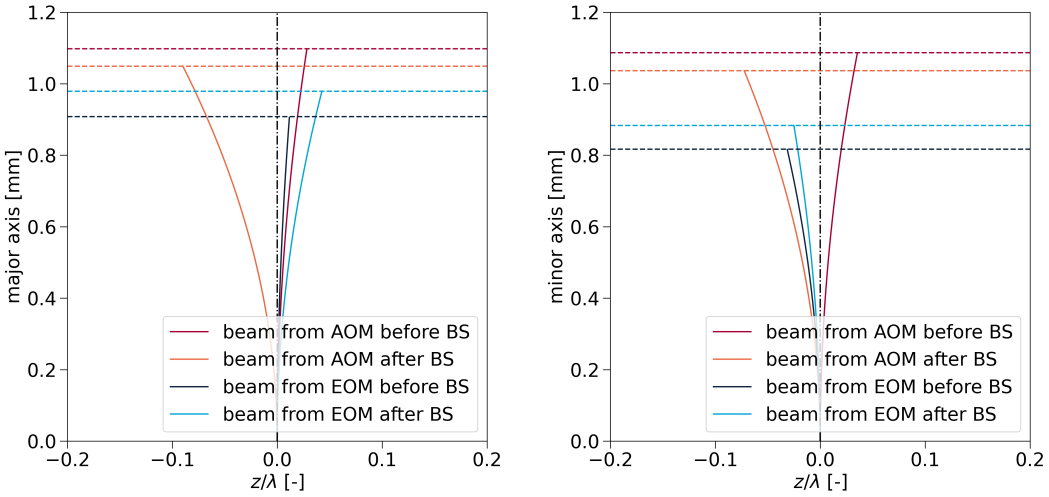
(a) major axis

(b) minor axis

Figure 3.11: The AOM (red) and EOM (blue) beam radii around the beam splitter (position indicated by the dash-dotted vertical line). To collimate the EOM beam a lens (position indicated by the dotted vertical line) is placed before the beam splitter. The position and focal length are determined with JamMt. The simulated beam after the lens is the blue dashed line. The major (a) and minor (b) axis are plotted separately. The different markers are used to differentiate the measured radii before and after the lens and beam splitter.

In this figure also the AOM beam (in red) before and after the beam splitter is plotted as well as the EOM beam after the beam splitter. This is to check that the AOM and EOM beams after the beam splitter behave similarly and that both waists are located near the beam splitter. Note that in figure 3.11 some of the measured radii are crossing the lens (dotted vertical line) and the beam splitter (dash-dotted vertical line) because they were measured before the lens and the beam splitter were placed. For both the AOM and EOM beam, the lines before and after the beam splitter do not line up at the beam splitter. This can be due to errors in determining the beam profiler positions and the use of different neutral density filters on the beam profiler.

To visualize the phase difference which may be introduced in the phase images by the curved wavefront the RoCs of the beams are calculated. Figure 3.12 shows the radius of curvature of the wavefront of the AOM (in red) and EOM (in blue) beam before (darker shade) and after (lighter shade) the beam splitter. The dotted lines are the beam radii and the vertical dash-dotted line is used as a reference. The horizontal axis is in terms of the wavelength. Therefore, the phase difference can be easily read as a percentage of 2π. The maximum phase difference along the major axis is approximately 0.18π and along the minor axis 0.15π. A more detailed discussion about the phase images and whether this phase difference is significant or not can be found in chapter 4.



(a) major axis

(b) minor axis

Figure 3.12: Radius of curvature of the wavefront in the major (a) and minor (b) axis direction of the AOM (red) and EOM (blue) beams before (darker shade) and after (lighter shade) the beam splitter. The dashed lines are the beam radii and the vertical dash-dotted line is shown as a reference at the beam splitter position. The horizontal axis is in terms of the wavelength to easily read the phase difference fraction. The vertical axis is added to quantify the beam sizes.

4

Phase imaging

The working principle of the phase camera has been discussed in the previous chapter. Now it is time to understand the phase images. First, the interference of two plane waves is discussed, after which the plane waves are exchanged for Gaussian waves. Here the theoretical background from chapter 2 is used to describe the Gaussian beam properties. Finally, the effect of the scanning mirror is taken into account which gives rise to an additional phase shift as a function of radial position from the beam axis. The mathematically derived phase differences introduced by the Gaussian beam optics and the scanning mirror are used to produce phase images via a simulation, which will be compared with real measurements.

4.1. Interference of two plane waves

Assume a beam splitter with two incoming plane waves as shown in figure 4.1. The beam splitter is placed under an angle of 45 degrees. The two incoming waves will interfere at the surface of a 50:50 beam splitter. Depending on their properties, the total power will be divided between the two output ports. Note that plane waves have a flat wavefront meaning that the phase is constant across a cross-section of the beam.

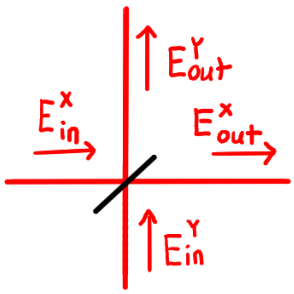


Figure 4.1: Beam splitter with two incoming beams and two interfered outgoing beams. Adapted from Van der Schaaf (2020).

4.1.1. Homodyne plane waves

First, assume that the two incoming waves have the same frequency. The incoming waves can then be described as $E_{in}^{x} = A_1 e^{i(\omega t - kz)}$ and $E_{in}^{y} = A_2 e^{i(\omega t - kz + \Delta\phi)}$ with A_1 and A_2 being real amplitudes, ω the frequency, t the time, k the wave number, z the coordinate in the direction of propagation and $\Delta\phi$

the phase difference between both waves. The outgoing intensities I_{out}^{x} and I_{out}^{y} are then given by

$$\begin{aligned} I_{out}^{x} = \left|E_{out}^{x}\right|^2 &= \left|\frac{A_1}{2}e^{i(\omega t - kz)} + \frac{A_2}{2}e^{i(\omega t - kz + \Delta\phi)}\right|^2 \\ &= \frac{1}{4}A_1^2 + \frac{1}{4}A_2^2 + \frac{1}{2}A_1A_2\mathrm{Re}\{-i\Delta\phi\} \\ &= \frac{1}{4}A_1^2 + \frac{1}{4}A_2^2 + \frac{1}{2}A_1A_2\cos(\Delta\phi)\,, \end{aligned} \tag{4.1}$$

and

$$\begin{aligned} I_{out}^{y} = \left|E_{out}^{y}\right|^2 &= \left|\frac{A_1}{2}e^{i(\omega t - kz + \pi)} + \frac{A_2}{2}e^{i(\omega t - kz + \Delta\phi)}\right|^2 \\ &= \frac{1}{4}A_1^2 + \frac{1}{4}A_2^2 + \frac{1}{2}A_1A_2\mathrm{Re}\{i\,(\pi - \Delta\phi)\} \\ &= \frac{1}{4}A_1^2 + \frac{1}{4}A_2^2 - \frac{1}{2}A_1A_2\cos(\Delta\phi)\,. \end{aligned} \tag{4.2}$$

Note the π phase shift in equation 4.2 which is introduced due to energy conservation at the beam splitter (Zetie et al., 2000). Also note that the intensity of the outgoing waves is constant, it only depends on the phase difference between the two incoming beams. Therefore, if a phase image would be taken of one of the outgoing beams, it would be constant throughout the beam spot.

4.1.2. Heterodyne plane waves

Next, the incoming waves are assumed to be heterodyne, i.e. they have different frequencies. They can be described as $E_{in}^{x} = A_1 e^{i(\omega_1 t - k_1 z)}$ and $E_{in}^{y} = A_2 e^{i(\omega_2 t - k_2 z + \Delta\phi)}$, where the subscripts are added to distinguish the frequency and wave number of both waves. The outgoing intensities can be calculated in the same way

$$\begin{aligned} I_{out}^{x} = \left|E_{out}^{x}\right|^2 &= \left|\frac{A_1}{2}e^{i(\omega_1 t - k_1 z)} + \frac{A_2}{2}e^{i(\omega_2 t - k_2 z + \Delta\phi)}\right|^2 \\ &= \frac{1}{4}A_1^2 + \frac{1}{4}A_2^2 + \frac{1}{2}A_1A_2\mathrm{Re}\{i\,([\omega_1 - \omega_2]\,t - [k_1 - k_2]\,z - \Delta\phi)\} \\ &= \frac{1}{4}A_1^2 + \frac{1}{4}A_2^2 + \frac{1}{2}A_1A_2\cos([\omega_1 - \omega_2]\,t - [k_1 - k_2]\,z - \Delta\phi)\,, \end{aligned} \tag{4.3}$$

and

$$\begin{aligned} I_{out}^{y} = \left|E_{out}^{y}\right|^2 &= \left|\frac{A_1}{2}e^{i(\omega_1 t - k_1 z + \pi)} + \frac{A_2}{2}e^{i(\omega_2 t - k_2 z + \Delta\phi)}\right|^2 \\ &= \frac{1}{4}A_1^2 + \frac{1}{4}A_2^2 + \frac{1}{2}A_1A_2\mathrm{Re}\{i\,(\pi + [\omega_1 - \omega_2]\,t - [k_1 - k_2]\,z - \Delta\phi)\} \\ &= \frac{1}{4}A_1^2 + \frac{1}{4}A_2^2 - \frac{1}{2}A_1A_2\cos([\omega_1 - \omega_2]\,t - [k_1 - k_2]\,z - \Delta\phi)\,. \end{aligned} \tag{4.4}$$

Now the singular intensity of the outgoing waves is not constant. It depends on where it is measured and at which time it is measured. The beat frequency with which the intensity fluctuates is $\omega_{beat} = \omega_1 - \omega_2$. A detector that is placed at a certain position behind the beam splitter will measure a flickering intensity with this frequency. If a phase measurement is conducted, the phase image will be constant over the beam spot. However, there can be a varying offset to the phase between different images due to the intensity fluctuations.

4.2. Interference of two Gaussian beams

In the previous section, the interference between two plane waves is discussed as well as the effect it has on the phase image. In reality, the measured beams are not plane waves, but Gaussian beams. Therefore, the curved wavefront and the Gouy phase of Gaussian beams have to be taken into account.

4.2.1. Curved wavefront

The curvature of the wavefront results in an extra phase difference to the outer edges of the beam spot with respect to a flat wavefront. Figure 4.2 shows a curved wavefront and the path length difference Δz with respect to a flat wavefront. A curved wavefront is defined by its radius of curvature $R\,(z)$ as described in equation 2.13.

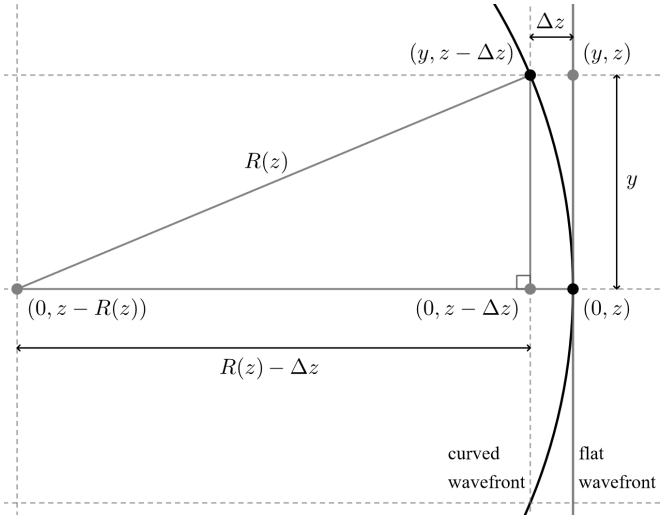


Figure 4.2: Schematic representation of the phase difference between a flat and a curved wavefront.

To find an expression for Δz, the Pythagorean theorem is used

$$[R\,(z)-\Delta z]^2 = R\,(z)^2 - y^2. \tag{4.5}$$

Solving this equation for Δz gives

$$\Delta z = R\,(z) - \sqrt{R\,(z)^2 - y^2}. \tag{4.6}$$

The beams in the actual set-up are conditioned such that they have a large radius of curvature at the beam splitter. Therefore, y is small compared to $R\,(z)$. Hence the square root can be approximated by

$$\sqrt{R\,(z)^2 - y^2} \approx R\,(z) - \frac{y^2}{2R\,(z)}. \tag{4.7}$$

Applying the approximation result in

$$\begin{aligned} \Delta z &= R\,(z) - \left[R\,(z) - \frac{y^2}{2R\,(z)}\right] \\ &= \frac{y^2}{2R\,(z)}. \end{aligned} \tag{4.8}$$

This path length difference can be converted into a phase difference

$$\Delta\phi\,(y,z) \approx \frac{2\pi}{\lambda}\frac{y^2}{2R\,(z)}, \tag{4.9}$$

where λ is the wavelength of the light. Note that this is exactly the second term of the phase of a lowest-order Gaussian beam as stated in equation 2.10, but for a one-dimensional case.

If the phase of a Gaussian beam is measured throughout a beam cross-section, the curved wavefront can be seen. The top right image in figure 2.4 shows the phase image of a lowest-order Gaussian beam ($\lambda = 1064$ nm). The radius of curvature in this example is approximately 6.36 m. Therefore, the

phase difference over the radius of the image ($y = 5$ mm) is approximately 3.69π. In the figure, the phase difference between the center and the edge is indeed of that order. Note that depending on the radius of curvature and beam spot size rings can be visible in the phase image. Small beam spots or large radii of curvature can look almost flat since the phase difference is small in these cases. Also, note that at the position of the waist and infinitely far away from the waist the Gaussian wavefront is flat. Hence, a Gaussian beam does not always have a phase difference depending on the radial coordinate.

The curved wavefront introduces an additional radial phase difference on top of the constant offset for a flat wavefront. This effect needs to be taken into account when studying the interference of two Gaussian beams. The phase camera studies the beat signal which is produced at the beam splitter. Therefore, the measured phase is the difference between the phase of the two incoming beams at the beam splitter. Neglecting the constant offset introduced by the flat wavefronts, this effect can be described as

$$\begin{aligned}\Delta\phi_{\textit{curved wavefront}}(y,z) &= \Delta\phi_1(y,z) - \Delta\phi_2(y,z)\\ &= \frac{2\pi}{\lambda_1}\frac{y^2}{2R_1(z_1)} - \frac{2\pi}{\lambda_2}\frac{y^2}{2R_2(z_2)}\\ &= \pi y^2\left[\frac{1}{\lambda_1 R_1(z_1)} - \frac{1}{\lambda_2 R_2(z_2)}\right].\end{aligned} \tag{4.10}$$

Note that z_1 and z_2 are a function of y since the beam splitter is placed under an angle with respect to the incoming beams. One side of the beam therefore later reaches the beam splitter then the other side. The radius of curvature therefore slightly differs for both sides. If the angle between the beam and the beam splitter is 45 degrees and $z = 0$ is at the center of the beam splitter, then $z = y$, resulting in

$$\Delta\phi_{\textit{curved wavefront}}(y) = \pi y^2\left[\frac{1}{\lambda_1 R_1(y)} - \frac{1}{\lambda_2 R_2(y)}\right]. \tag{4.11}$$

4.2.2. Gouy phase

Another contribution to the phase of a Gaussian beam is the Gouy phase. This describes an extra longitudinal phase lag originating from the slower phase velocity of a Gaussian beam compared to a plane wave (Freise & Strain, 2010). The Gouy phase for the lowest-order mode equals

$$\Phi(z) = \arctan\left(\frac{z - z_0}{z_R}\right), \tag{4.12}$$

with z_0 being the waist position and z_R the Rayleigh range. Note that the total phase lag from $z = -\infty$ to $z = \infty$ equals π. Note that the Gouy phase depends only on the z coordinate, which is in the direction of propagation of the beams. Therefore, the Gouy phase is a constant offset to the phase difference like the effect of plane waves. This offset can be calculated by taking the difference between the Gouy phases of the incoming beams

$$\begin{aligned}\Delta\phi_{\textit{Gouy phase}}(z) &= \Phi_1(z) - \Phi_2(z)\\ &= \arctan\left(\frac{z_1 - z_{0,1}}{z_{R,1}}\right) - \arctan\left(\frac{z_2 - z_{0,2}}{z_{R,2}}\right).\end{aligned} \tag{4.13}$$

Since the Gouy phase introduces only an offset, the effect can not be seen in one image. However, it does contribute to the overall phase offset.

4.3. Interference with a scanning mirror

In the prototype set-up of the phase camera, the scanner determines which part of the wavefront that we want to probe reaches the photodiode. The angle of the scanner mirror changes the optical path length for different image pixels which introduces an extra phase difference. Therefore, it is necessary to calculate the path lengths for different angles of the mirror in the scanner.

A sketch of the set-up for two different scanner angles is drawn in figure 4.3. Note that the figure is not to scale since the path lengths are much larger than the beam widths in reality. The scanner mirror is drawn in the central position where the incoming angle of the beam is $\frac{\pi}{4}$, and in the positions where it is rotated by a small rotation α from the central position. The light rays for these scanner positions that reach the photodiode box are drawn as well. For the central position of the mirror, the center of the beam reaches the photodiode. For a mirror angle α, light that is not on the propagation axis of the beam is directed to the photodiode. This is indicated by a second ray displaced by a distance Δy. Between the scanner and the photodiode box a beam splitter is placed where the test and reference beam can interfere, thereby creating the beat signal. Two light rays of the reference beam coming from the AOM are drawn too, exactly at the positions where the light rays coming from the scanner hit the beam splitter. Since only the additional phase difference needs to be calculated all wavefronts are assumed flat, in the end, all contributions are added for the total effect.

For clarity, multiple distances are defined in this figure as well as some angles. Note the difference between the single and double-lined red angles which equal an angle of α and 2α, respectively. The other two defined angles are the blue angle of $\frac{\pi}{4}$ and the green angle being $\frac{\pi}{4} - \alpha$. The inset of this figure shows a simplified representation of the studied scanner configuration.

To calculate the total phase difference between the two light rays of the test beam due to the path length differences it is convenient to split the path in several parts. Starting with the length difference up to the scanner ΔL_1, this difference is defined as

$$\Delta L_1 = L_1' - L_1 = \Delta y \tan\left(\frac{\pi}{4} - \alpha\right). \tag{4.14}$$

The next step is to calculate the total length L_2' from the tilted mirror to the photodiode. Using the Pythagorean theorem L_2' can be expressed in terms of L_2 and Δy:

$$L_2' = \sqrt{{\Delta L_1}^2 + (L_2 - \Delta y)^2}. \tag{4.15}$$

To write L_2' as a function of only Δy and α, the next step is to find an expression for L_2 in terms of Δy. Therefore, the tangent function is used in the same triangle

$$\tan(2\alpha) = \frac{\Delta L_1}{L_2 - \Delta y}. \tag{4.16}$$

Rewriting and substituting equation 4.14 into the previous equation gives

$$L_2 - \Delta y = \frac{\tan\left(\frac{\pi}{4} - \alpha\right)}{\tan(2\alpha)} \Delta y. \tag{4.17}$$

Now solving for L_2 by using the shifted half-angle formula for the tangent results in

$$\begin{aligned} L_2 &= \left[\frac{\tan\left(\frac{\pi}{4} - \alpha\right)}{\tan(2\alpha)} + 1\right] \Delta y \\ &= \frac{\Delta y}{\sin(2\alpha)}. \end{aligned} \tag{4.18}$$

Substituting equations 4.14 and 4.18 into equation 4.15 shows that L_2' depends only on Δy and α

$$\begin{aligned} L_2' &= \sqrt{\left(\Delta y \tan\left(\frac{\pi}{4} - \alpha\right)\right)^2 + \left(\frac{\Delta y}{\sin(2\alpha)} - \Delta y\right)^2} \\ &= \Delta y \sqrt{\left(\tan\left(\frac{\pi}{4} - \alpha\right)\right)^2 + \left(\frac{1}{\sin(2\alpha)} - 1\right)^2} \\ &= \frac{\tan\left(\frac{\pi}{4} - \alpha\right)}{\sin(2\alpha)} \Delta y. \end{aligned} \tag{4.19}$$

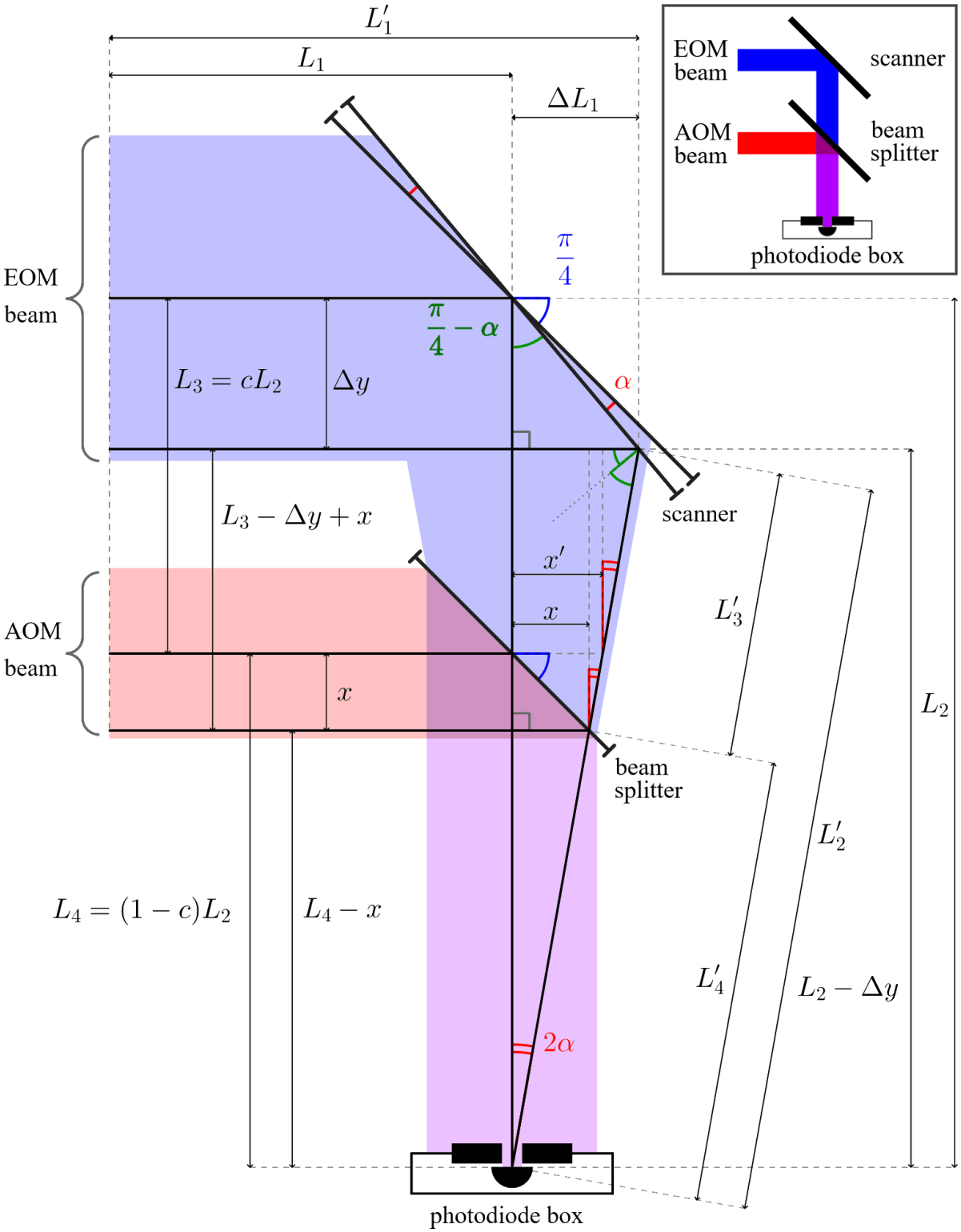


Figure 4.3: Schematic representation of the optical set-up with two positions of the scanner. Two light rays are drawn which reach the photodiode box to calculate all path optical path lengths. Note that the image is not to scale since the beam widths are much smaller than the optical path lengths in reality.

Now taking the difference between L_2' and L_2 gives

$$\begin{aligned}\Delta L_2 &= L_2' - L_2 \\ &= \frac{\tan\left(\frac{\pi}{4}-\alpha\right)}{\sin(2\alpha)}\Delta y - \frac{1}{\sin(2\alpha)}\Delta y \\ &= \frac{\tan\left(\frac{\pi}{4}-\alpha\right)-1}{\sin(2\alpha)}\Delta y.\end{aligned} \tag{4.20}$$

Next, L_2 is split into L_3 and L_4 which are the distance between the scanner and the beam splitter and the beam splitter and the photodiode box, respectively. The constant c is defined as the fraction of L_3 and L_2 such that $L_3 = cL_2$ and $L_4 = (1-c)\,L_2$. Before L_3 and L_4 can be calculated, the position of the beams on the beam splitter must be known. Therefore, the distance x, the distance between the two parallel incoming reference beam light rays at the beam splitter, has to be derived. To start, it is noted that x' equals

$$x' = x\,[1+\tan(2\alpha)]\,. \tag{4.21}$$

Now taking the tangent in the triangle spanning from the photodiode up to x' gives

$$\begin{aligned}\tan(2\alpha) &= \frac{x'}{L_4} \\ &= \frac{x\,[1+\tan(2\alpha)]}{(1-c)\,L_2} \\ &= \frac{x\,[1+\tan(2\alpha)]\sin(2\alpha)}{(1-c)\,\Delta y} \\ &= \frac{x\tan(2\alpha)\,[\sin(2\alpha)+\cos(2\alpha)]}{(1-c)\,\Delta y}.\end{aligned} \tag{4.22}$$

To find x, the above equation can be rewritten into

$$x = \frac{1}{\sin(2\alpha)+\cos(2\alpha)}\,(1-c)\,\Delta y. \tag{4.23}$$

Now L_3' can be calculated by taking a cosine in the triangle from the scanner up to the beam splitter

$$\begin{aligned}L_3' &= \frac{cL_2 - \Delta y + x}{\cos(2\alpha)} \\ &= \frac{\frac{c}{\sin(2\alpha)}\Delta y - \Delta y + \frac{1}{\sin(2\alpha)+\cos(2\alpha)}\,(1-c)\,\Delta y}{\cos(2\alpha)} \\ &= \frac{\frac{1}{\sin(2\alpha)}c - 1 + \frac{1}{\sin(2\alpha)+\cos(2\alpha)}\,(1-c)}{\cos(2\alpha)}\Delta y \\ &= \left[\frac{\frac{1}{\sin(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)}c + \frac{\frac{1}{\cos(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)} - \frac{1}{\cos(2\alpha)}\right]\Delta y.\end{aligned} \tag{4.24}$$

Next, the path length difference ΔL_3 can be calculated

$$\begin{aligned}\Delta L_3 &= L_3' - L_3 \\ &= \left[\frac{\frac{1}{\sin(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)}c + \frac{\frac{1}{\cos(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)} - \frac{1}{\cos(2\alpha)} - \frac{c}{\sin(2\alpha)}\right]\Delta y \\ &= \left(\left[\frac{\frac{1}{\sin(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)} - \frac{1}{\sin(2\alpha)}\right]c + \left[\frac{\frac{1}{\cos(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)} - \frac{1}{\cos(2\alpha)}\right]\right)\Delta y.\end{aligned} \tag{4.25}$$

The last path length that needs to be calculated is L_4' and its deviation from L_4, ΔL_4. Therefore, the bottom triangle is used. First, the side $L_4 - x$ is defined

$$\begin{aligned} L_4 - x &= (1-c)\,L_2 - x \\ &= \frac{1}{\sin(2\alpha)}(1-c)\,\Delta y - \frac{1}{\sin(2\alpha)+\cos(2\alpha)}(1-c)\,\Delta y \\ &= \frac{\frac{1}{\tan(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)}(1-c)\,\Delta y. \end{aligned} \tag{4.26}$$

To calculate L_4', the cosine is applied to the same triangle

$$\begin{aligned} L_4' &= \frac{L_4 - x}{\cos(2\alpha)} \\ &= \frac{\frac{1}{\sin(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)}(1-c)\,\Delta y. \end{aligned} \tag{4.27}$$

Lastly, the difference ΔL_4 can be calculated

$$\begin{aligned} \Delta L_4 &= L_4' - L_4 \\ &= \frac{\frac{1}{\sin(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)}(1-c)\,\Delta y - \frac{1}{\sin(2\alpha)}(1-c)\,\Delta y \\ &= \left[\frac{\frac{1}{\sin(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)} - \frac{1}{\sin(2\alpha)}\right](1-c)\,\Delta y. \end{aligned} \tag{4.28}$$

A difference in optical path lengths for different light rays results in a phase difference equal to

$$\Delta\phi = \frac{2\pi}{\lambda}\Delta L. \tag{4.29}$$

The phase camera measures the beat signal which is produced by the interference at the beam splitter. Therefore, the only relevant phase behind the beam splitter is the phase of the beat. The beat signal has a frequency of $f_{beat} = 80$ MHz which corresponds with a beat wavelength of $\lambda_{beat} \approx 3.75$ m. This wavelength is much larger than that of the beams coming from the EOM and AOM. The wavelength of the carrier signal of the EOM beam equals $\lambda_{EOM,car} = 1064$ nm while the wavelength of the AOM equals $\lambda_{AOM} \approx 1064$ nm-300 fm due to the 80 MHz frequency shift. The phase difference due to a path length difference is inversely related to the wavelength. Therefore, the path length difference before the beam splitter dominates the effect. The total effect of the path length difference can be calculated with

$$\Delta\phi_{\textit{path length}}(\Delta y) = 2\pi\left[\frac{\Delta L_{EOM}(\Delta y)}{\lambda_{EOM}} - \frac{\Delta L_{AOM}(\Delta y)}{\lambda_{AOM}} + \frac{\Delta L_{beat}(\Delta y)}{\lambda_{beat}}\right]. \tag{4.30}$$

Note the minus sign before the second term. The phase of the beat signal is the difference between the phase of the EOM and AOM beam that interfere at the beam splitter. The latter term is very small compared to the first two terms because the beat frequency is small compared to the light frequencies.

Since the wavelength differs on each side of the beam splitter, the path lengths have to be calculated for each section. Two light rays coming from the EOM have a path length difference of

$$\begin{aligned} \Delta L_{EOM} &= \Delta L_1 + \Delta L_3 \\ &= \left(\left[\frac{\frac{1}{\sin(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)} - \frac{1}{\sin(2\alpha)}\right]c + \left[\frac{\frac{1}{\cos(2\alpha)}}{\sin(2\alpha)+\cos(2\alpha)} - \frac{1}{\cos(2\alpha)} + \tan\left(\frac{\pi}{4}-\alpha\right)\right]\right)\Delta y. \end{aligned} \tag{4.31}$$

The path length difference for the two rays from the AOM is

$$\begin{aligned}\Delta L_{AOM} &= x \\ &= \frac{1}{\sin(2\alpha) + \cos(2\alpha)} (1-c)\,\Delta y,\end{aligned} \tag{4.32}$$

and the path length difference from the beam splitter up to the photodiode box equals

$$\begin{aligned}\Delta L_{beat} &= \Delta L_4 \\ &= \left[\frac{\frac{1}{\sin(2\alpha)}}{\sin(2\alpha) + \cos(2\alpha)} - \frac{1}{\sin(2\alpha)}\right](1-c)\,\Delta y.\end{aligned} \tag{4.33}$$

Equation 4.30 combines these three path length differences into a total radial phase difference.

4.4. Interference in the phase camera set-up

To describe the total phase images of the phase camera, all previous effects need to be combined. However, the fact that the phase camera uses heterodyne detection introduces another level of complexity since it acquires data at multiple frequencies. Therefore, the total phase effect for a single frequency is determined, after which its frequency dependence is discussed.

4.4.1. Total phase

The several effects which introduce phase differences in the phase camera set-up have been discussed in the previous sections. They can be sorted into two groups: radial effects and total offsets. The radial effects can influence single phase images while the offset effects only are visible between images taken at different times. Here, only the radial phase differences are studied.

Two effects contribute to the radial phase differences: the path length difference and the curved wavefront. The total radial phase difference can be calculated by adding these two terms

$$\begin{aligned}\Delta\phi &= \Delta\phi_{\textit{path length}} + \Delta\phi_{\textit{curved wavefront}} \\ &= 2\pi\left[\frac{\Delta L_{EOM}}{\lambda_{EOM}} - \frac{\Delta L_{AOM}}{\lambda_{AOM}} + \frac{\Delta L_{beat}}{\lambda_{beat}}\right] + \\ &\quad \pi x^2\left[\frac{1}{\lambda_{EOM} R_{EOM}(z_{EOM})} - \frac{1}{\lambda_{AOM} R_{AOM}(z_{AOM})}\right].\end{aligned} \tag{4.34}$$

The radial coordinate y is replaced by x in the curved wavefront phase difference because the definitions from figure 4.3 are used. To calculate the radii of curvature of the AOM and EOM beams, the location of the beam splitter needs to be known along the beam axis. The center of the beam splitter is taken as $z = 0$. The position where the light reaches the beam splitter is then the path length difference ΔL_{AOM} as derived in section 4.3. Therefore, $z_{AOM} = \Delta L_{AOM} - z_{0,AOM}$, with $z_{0,AOM}$ the position of the waist of the AOM beam. For the EOM beam, the path length difference is $z_{EOM} = \Delta L_{EOM} - z_{0,EOM}$, with $z_{0,EOM}$ the position of the waist of the EOM beam. This results in a total radial phase difference

$$\begin{aligned}\Delta\phi(\Delta y) = 2\pi&\left[\frac{\Delta L_{EOM}}{\lambda_{EOM}} - \frac{\Delta L_{AOM}}{\lambda_{AOM}} + \frac{\Delta L_{beat}}{\lambda_{beat}}\right] + \\ &\pi x^2\left[\frac{1}{\lambda_{EOM} R_{EOM}\left(\Delta L_{EOM} - z_{0,EOM}\right)} - \frac{1}{\lambda_{AOM} R_{AOM}\left(\Delta L_{AOM} - z_{0,AOM}\right)}\right].\end{aligned} \tag{4.35}$$

This is exclusively a function of Δy since all the path length differences only depend on Δy.

4.4.2. Sidebands

In the previous sections, only the wavelength of the carrier frequency of the EOM is mentioned. However, the EOM creates sidebands as described in section 3.2.2. For completeness, the sideband frequencies of the EOM are described in this section.

In the prototype set-up, the wavelength of the laser equals $\lambda_{laser} = 1064$ nm. The EOM creates sidebands at frequencies $f_{EOM} = f_{laser} \pm n_{sb}\Delta f_{EOM}$, with f_{laser} the laser frequency, n_{sb} the number of the upper/lower sideband and $\Delta f_{EOM} = 5$ MHz the frequency difference created by to the EOM. The light coming from the AOM on the other hand has a frequency shift $\Delta f_{AOM} = 80$ MHz compared to the laser, $f_{AOM} = f_{laser} + \Delta f_{AOM}$.

After the interference at the beam splitter, the beat signal is measured at the photodiode. The studied signals in the demodulation are the carrier (car) signal and the first upper (USB) and lower (LSB) sideband signal, hence $n_{sb} = 1$. These represent the following frequencies of the light coming from the EOM

$$\begin{aligned} f_{EOM,car} &= f_{laser}, \\ f_{EOM,USB} &= f_{laser} + \Delta f_{EOM}, \\ f_{EOM,LSB} &= f_{laser} - \Delta f_{EOM}. \end{aligned} \tag{4.36}$$

Note that the studied beat frequencies are the difference between the light coming from the AOM and EOM. Therefore, the beat frequencies equal

$$\begin{aligned} f_{beat,car} &= \Delta f_{AOM}, \\ f_{beat,USB} &= \Delta f_{AOM} - \Delta f_{EOM}, \\ f_{beat,LSB} &= \Delta f_{AOM} + \Delta f_{EOM}. \end{aligned} \tag{4.37}$$

To quantize the phase difference induced by the path length difference, the different wavelengths need to be known. These can be calculated from the frequencies in equation 4.36

$$\begin{aligned} \lambda_{EOM,car} &= 1064 \text{ nm}, \\ \lambda_{EOM,USB} &= 1064 \text{ nm} - 20 \text{ fm}, \\ \lambda_{EOM,LSB} &= 1064 \text{ nm} + 20 \text{ fm}. \end{aligned} \tag{4.38}$$

The difference in the wavelength of the light of the carrier and first sidebands is approximately 20 fm. This is eight orders smaller than the wavelength of the laser and can therefore be neglected in the calculation of the radial phase differences.

In the beat signal, the frequency of the sidebands is approximately 6% off compared to the carrier signal. However, a phase difference between carrier and sideband signals behind the beam splitter can be neglected since the path length difference before the beam splitter is more than six orders larger than the path length difference behind the beam splitter.

The combination of the last two statements means that it is expected that the phase images of the carrier, USB and LSB signal are the same except for a possible offset. This can indeed be seen in measurements, for example in figure 3.9. The phase images are all similar except for an offset and noise. The sidebands have a lower intensity than the carrier signal which makes them more sensitive to noise.

4.5. Simulation of the phase images

The total radial phase difference which influences the phase images is described in equation 4.35. This phase difference is derived in one dimension. To create a phase image from the one-dimensional phase difference, Δy needs to be varied from 0 to the maximum scan radius. A two-dimensional image is made by rotation since the phase image is rotationally symmetric. A Python script has been written to simulate the phase images from the mathematical phase difference derived in the previous section. The images are created on a grid of 501 by 501 pixels to retrieve sharp images. The simulation takes approximately 7.0 s to create the wavefront images. The Python code can be found in appendix B.

An example of a simulated phase image is shown in figure 4.4. The prototype set-up parameters are used such that the distance between the scanner and photodiode $L_2 = 85$ cm and $L_3 = 22.5$ cm. The EOM and AOM are profiled and the fitted beam parameters are used to determine their radii of curvature. The AOM frequency is $\Delta f_{AOM} = 80$ MHz and the image is scanned over with a maximum scan radius of 2 mm. The phase differences at the carrier frequency are simulated. In the phase image rings are visible with constant phase. Note that the width of these rings decreases the further they are away from the center since the phase difference rapidly increases with increasing scan radius. In total 3.5 rings are visible.

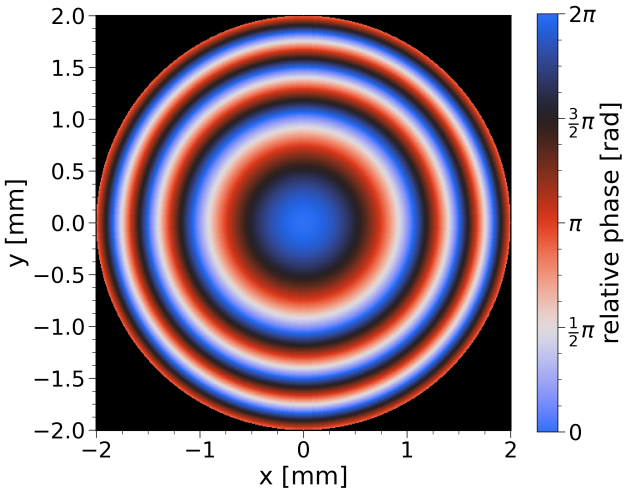


Figure 4.4: Simulated phase image with the prototype set-up parameters.

The relation between $\Delta\phi$ and Δy is shown in figure 4.5 as the black line. The blue line and the red line are the contributions due to the path length difference and curved wavefront, respectively. On the right axis, the phase difference is divided by 2π to show how often the phase wraps, i.e., the number of rings of the same color the phase image will have. It can be seen that the phase difference rapidly increases with increasing scan radius Δy. Therefore, the ring width will decrease with increasing scan radius. Note that the contribution of the path length difference to phase difference is dominant. The contribution of the curved wavefront is small and in this case, partly compensates for the path length difference. If the beam would be converging instead of diverging, then this contribution would add a small extra phase difference.

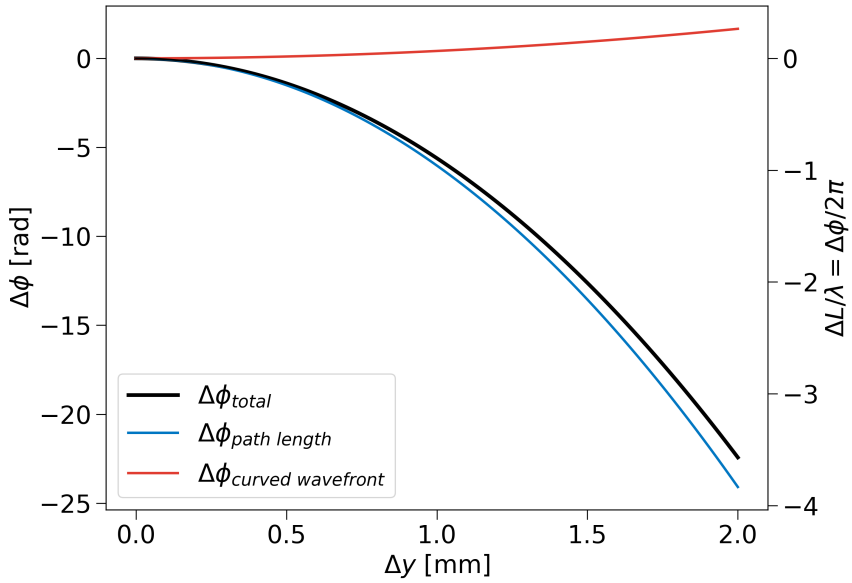


Figure 4.5: Theoretical phase difference $\Delta\phi$ plotted versus the scan radius Δy. On the right axis, the phase difference has been divided by 2π to see how often the phase wraps.

The distance between the rings varies as stated earlier. The size of a ring is measured by taking the difference between the radius Δy at two consecutive phase values where $\Delta\phi/2\pi = 0$. Figure 4.6 shows the ring size as a function of the ring number for the same simulation as before but now with a scan radius of 4 mm to see the width of more rings. The first ring is much wider than the other rings.

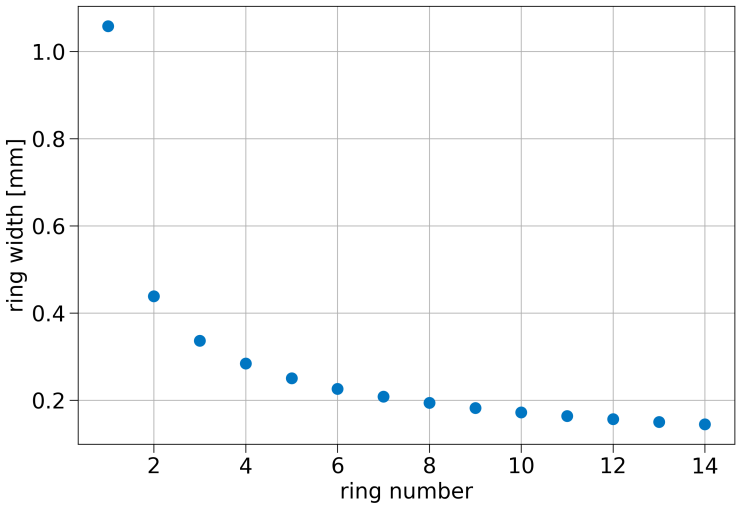


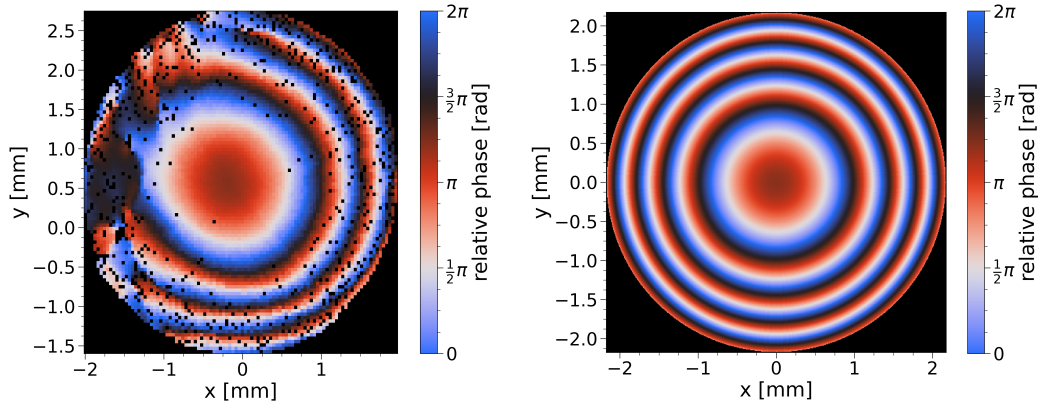
Figure 4.6: Ring width versus the ring number.

4.6. Validation of the simulation

To validate the simulation, results are compared with phase images taken with the prototype set-up. In this section, a single measurement is compared with the simulation, after which the measurement method for the number of rings is explained. Finally, the relation between the number of rings and the photodiode position is discussed.

4.6.1. Single measurement

In figure 4.7, the phase image of a measurement of the carrier signal is shown on the left side. The photodiode was placed at $L_2 = 85$ cm and the beam splitter and $L_3 = 22.5$ cm from the scanner. The image has a scan radius of 1.99 mm in the x-direction and 2.17 mm in the y-direction. This difference can be explained by a calibration error of the scanner. Due to the scanning pattern, there are black pixels that do not contain data points. On the right side, the simulated phase image is shown with a scan radius of 2.17 mm. The simulated phase difference due to a curved wavefront is based on measured beam parameters of the AOM and EOM beam. Along the vertical axis, the measured phase image shows approximately 3.63 rings. While the simulation predicts 4.04 rings, which is 11% off. The horizontal direction measures 2.96 rings while the simulation predicts 3.57. This is a difference of 21%. This difference between the horizontal and vertical axis in the measured image can be due to aberrations in the optical set-up. Note that the exact position of the photodiode in the photodiode box is unknown. It is assumed at a depth of 1 cm from the front side of the box. This introduces an uncertainty in L_2 which can partly explain the difference between the measurement and the simulation.



(a) measured wavefront

(b) simulated wavefront

Figure 4.7: Comparison of a measured (a) and simulated (b) wavefront of the phase camera. The number of rings in the simulation is 21% larger compared to the measured image in the horizontal direction and 11% in the vertical direction.

4.6.2. Measurement method for the number of phase rings

To determine the number of rings in a measurement, the phase images should be centered within the scanned area. The shift is created by adding a voltage offset to the signal that drives the scanning mirror. In the example measurement in figure 4.8, the images are shifted with $\Delta x = 0.5$ mm and $\Delta y = -1.5$ mm with respect to the intensity-centered images. Since this shift steers the center of the beam away from the photodiode, the intensity of the edges of half the image decreases significantly. Therefore, more noise is visible in the left top of the example measurement. From the phase image of the carrier signal, half of the column that contains the center of the phase image is taken manually. This half column is marked in yellow in figure 4.8.

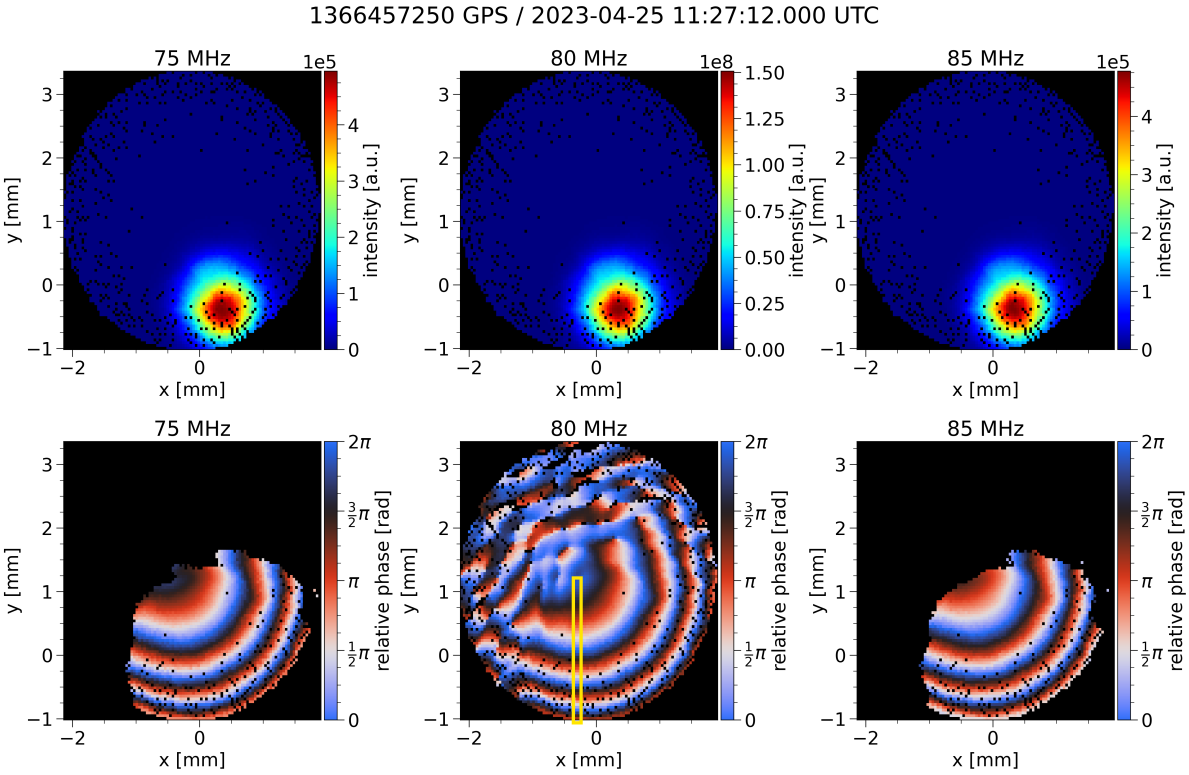


Figure 4.8: Example measurement of the phase camera with $L_2 = 0.85$ cm to determine the number of phase rings. The phase image is centered to measure the phase difference along the scanning radius. The yellow box shows the half column which is used to count the number of rings.

Next, the phase values from this half column are plotted versus the scan radius. The measured phase is wrapped to the interval of 0 to 2π. Figure 4.9 shows the wrapped phase as blue dots. To compare it with the simulation, the measured phase has been unwrapped. For the black pixels in the yellow column, the data is linearly interpolated to regain the unwrapped relation. The red squares in this image show the unwrapped phase. To calculate the total phase difference across the scanning radius of the image, the difference between the first and last unwrapped phase values is taken. This value is divided by 2π to get the number of rings along the scan radius. Since not all measurements have the same scanning radius, the difference between the radial coordinates is taken to determine the scan radius.

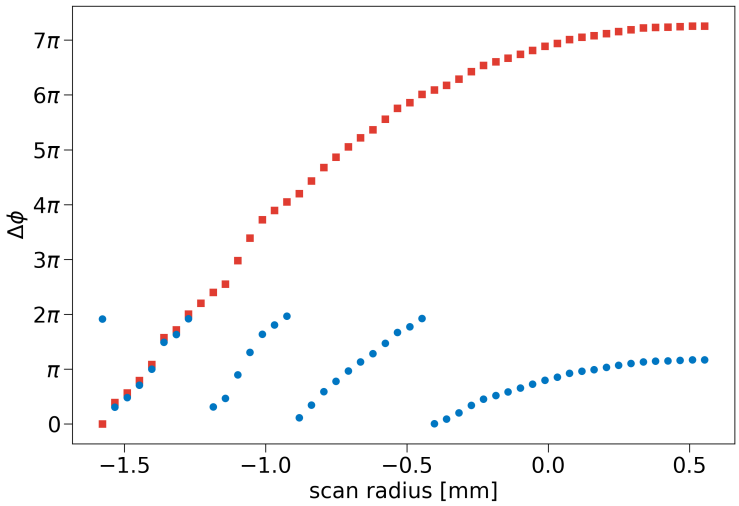


Figure 4.9: Phase difference versus the scan radius. The blue dots are the measured values from the half column, which are wrapped. The red squares show the unwrapped phase.

4.6.3. Number of phase rings versus photodiode position

Finally, the relation between the number of rings visible and the photodiode position is determined. The position of the photodiode is varied and multiple images are taken (see figures C.1 to C.12). The photodiode has been moved from $L_2 = 0.30$ cm up to $L_2 = 0.85$ cm in steps of 5 cm while the final beam splitter is kept at $L_3 = 22.5$ cm. Note that this means that c varies too. The scanning radius is kept constant at 2 mm by altering the amplitude of the DC input voltages of the scanner for each measurement. In appendix C the measurements are shown, which are used to measure the number of rings as described in section 4.6.2. Note that the intensity images are not centered and the phase images are a bit blurred since the photodiode is moved. The positioning of the photodiode strongly influences the alignment because the beams are focused on its active area with a diameter of 55 μm. The simulation predicts that the number of rings decreases with increasing L_2 as shown by the red line in figure 4.10. In this figure, the measured number of rings is plotted as blue dots. Note that they also decrease with increasing L_2, but with a different trend.

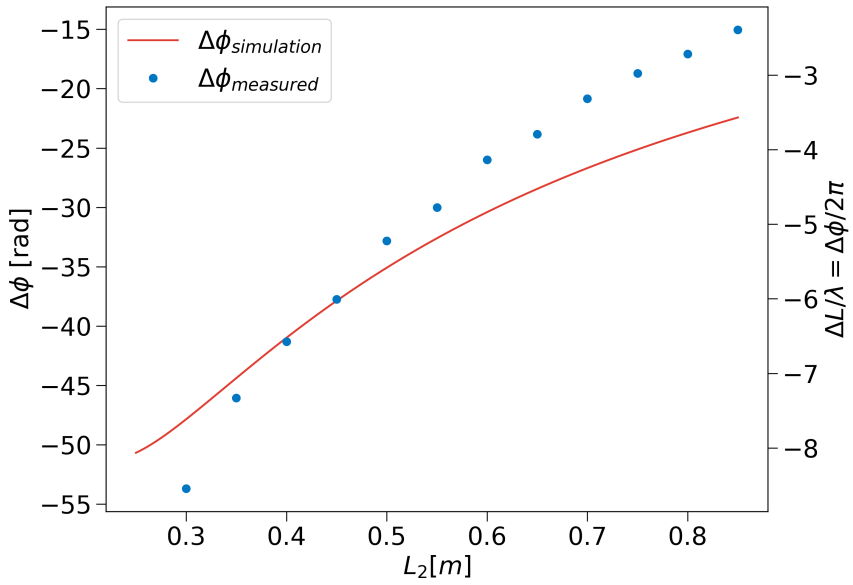


Figure 4.10: Simulated phase difference relation (red line) at a scanning radius of 2 mm for varying photodiode positions L_2. The final beam splitter where the test and reference beam recombine is kept at $L_3 = 22.5$ cm. The blue dots are phase differences measured with the prototype set-up.

5

Phase stability

The prototype set-up at Nikhef is built to better understand the phase camera in a controlled environment. In this chapter, measurements on the phase stability of the phase camera are presented to characterize it. First, the stability over long timescales is discussed to see low-frequency fluctuations between measurements. Next, the high-frequency noise on phase measurements is measured to quantify the phase resolution for different test and reference powers. These measurements are compared with expected noise levels from Van der Schaaf (2020).

5.1. Long timescale phase stability

With the current prototype set-up, the relative phase is measured between the beat signal and the signals of the generators powering the AOM and EOM. An absolute phase measurement is currently not possible, because the points are measured sequentially (Van der Schaaf, 2020). However, it would be interesting to know to what extent an absolute phase measurement is possible. The stability of relative phase measurement needs to be studied since the phase camera is based on an interferometric measurement that has a high sensitivity. Small disturbances such as airflow can influence the results significantly. To measure the long timescale stability, tens of seconds, of the relative phase, the scanning mirror is placed at a stationary angle. The phase of the carrier signal is measured for 100 periods of approximately 0.5 seconds. These periods of 0.5 seconds are chosen because then the DAQ of the phase camera can be used. The measured phase difference between the photodiode and the generator signals is plotted over time in figure 5.1.

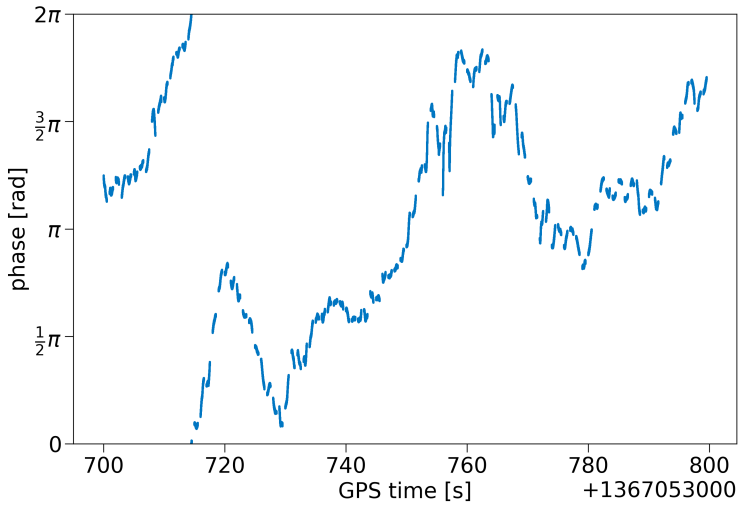


Figure 5.1: Stability measurement of the phase difference between the carrier signal and the reference signal from the generators powering the AOM and EOM. The set-up was not covered during this measurement.

If there would be no phase noise, then the measured relative phase would be constant over time. However, there are low-frequency fluctuations all over a range of almost 3π, and the largest fluctuation in one interval of 0.5 seconds is approximately 0.4π. This means that within one phase image, there can be phase fluctuations between measured points of that order. Note a path length difference in the two long paths between BS1 and BS3 (1 to 1.5 m) in figure 3.2 of 1.6 μm can cause this 3π phase shift since the laser wavelength is 1064 nm. Therefore, airflow in the set-up could be partly responsible for these fluctuations.

To investigate this effect, most of the set-up is covered to reduce disturbances due to the airflow from the air-conditioning. The set-up was only partly covered due to the limitation of materials. To prevent airflow from one side to the other, a small strip on one side instead of both sides of the set-up was left uncovered. To quantify the effect, another stability measurement was conducted. The result of this second measurement can be seen in figure 5.2. Note that the phase stability significantly improved. In the previous measurement, the phase fluctuated over a range of almost 3π, while now the phase only fluctuates over a domain of 0.75π. There is a maximum fluctuation of 0.2π within one interval of 0.5 seconds.

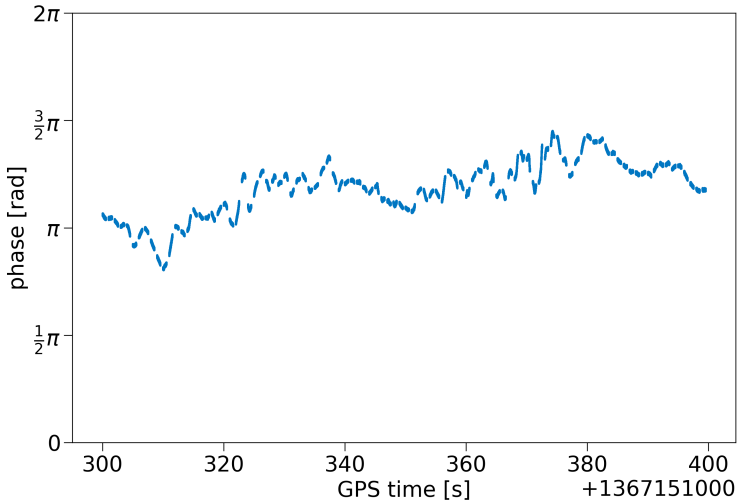


Figure 5.2: Stability measurement of the phase difference between the carrier signal and the reference signal from the generators powering the AOM and EOM. Most of the set-up was covered during this measurement.

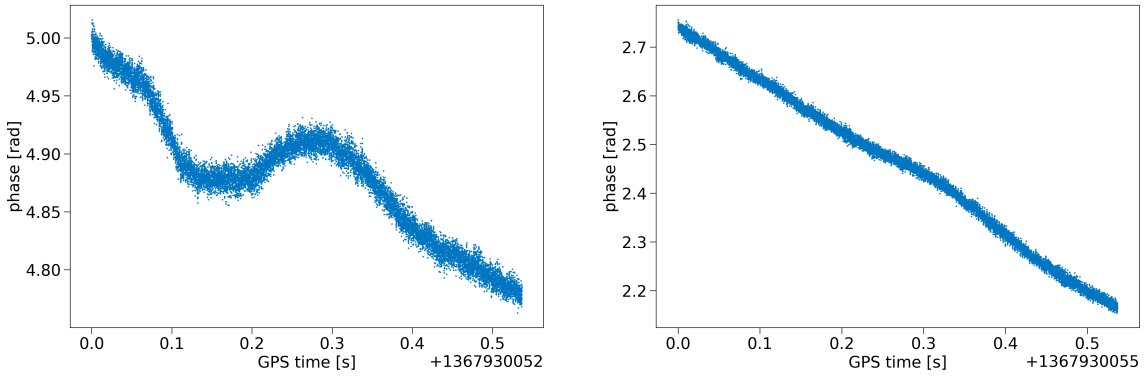
This significant increase in stability by partly covering the set-up is hopeful. When possible, the whole set-up should be covered in the future to protect the set-up against airflow and to stabilize temperature. Note that with the current set-up, one cannot compensate for these fluctuations, because this measurement cannot be performed simultaneously with a normal scanning phase camera measurement.

Possible other effects that influence the phase stability are polarization fluctuations and the instability of the AOM. These are further discussed in chapter 6.

5.2. Phase resolution

In the previous section the long timescale, tens of seconds, fluctuations are discussed. These low-frequency fluctuations are significant and vary strongly from interval to interval. Here, the high-frequency point-to-point fluctuations on timescales of 10 to 100 ms are studied. This phase noise, compared to the low-frequency fluctuations, is unlikely to be temperature related since it happens on small timescales. Two example measurements of the phase of the carrier frequency in different intervals are shown in figure 5.3. The left one has a lot of low-frequency fluctuations, while the right one shows a more linearly decreasing behavior. Note that the right one is also not constant over time as desired. However, the high-frequency noise on the phase measurement in both measurements is of the same order. The width of both lines is in the order of 25 mrad. These fast fluctuations are considered the phase reso-

lution. In the coming section, a measurement method for the phase resolution is explained. Next, the phase resolution is measured for varying beam powers and compared with theory.



(a) example measurement, taken at 1367930052 s (GPS time)

(b) example measurement, taken at 1367930055 s (GPS time)

Figure 5.3: Two example measurements of the relative phase in one time interval of 0.5 s. The low-frequency fluctuations between both measurements are different while the high-frequency noise on the phase measurements (width of the lines) is of the same order, 25 mrad.

5.2.1. Measurement method

The slow trends of the low-frequency fluctuations can be separated from the high-frequency point-to-point fluctuations. To calculate the phase resolution from these measurements, the intervals with linearly looking segments such as figure 5.3b are selected. The linear segments of these intervals are taken apart and a linear function is fitted to them with a least squared optimization. Figure 5.4 shows the linear segment of the first 0.25 s of one of the example measurements. The red line is fitted linear function.

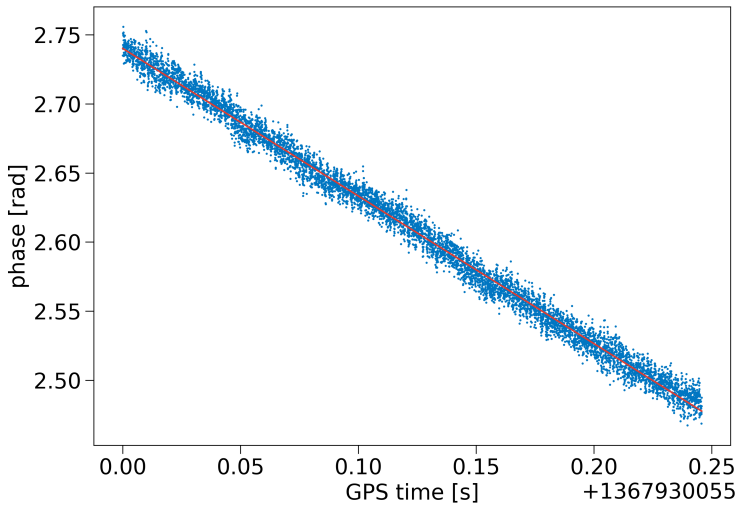


Figure 5.4: Linear segment of the first 0.25 s of the measurement in figure 5.3b. The red line is a fitted linear function.

Next, the difference between the data points and the linear function is calculated. A histogram of this difference is shown in figure 5.5 for 7500 phase measurements. Note that it is approximately a normal distribution. To determine the resolution of the phase measurements, the standard deviation is calculated. For this example, the standard deviation and thus the phase resolution is $\sigma = 6.79$ mrad.

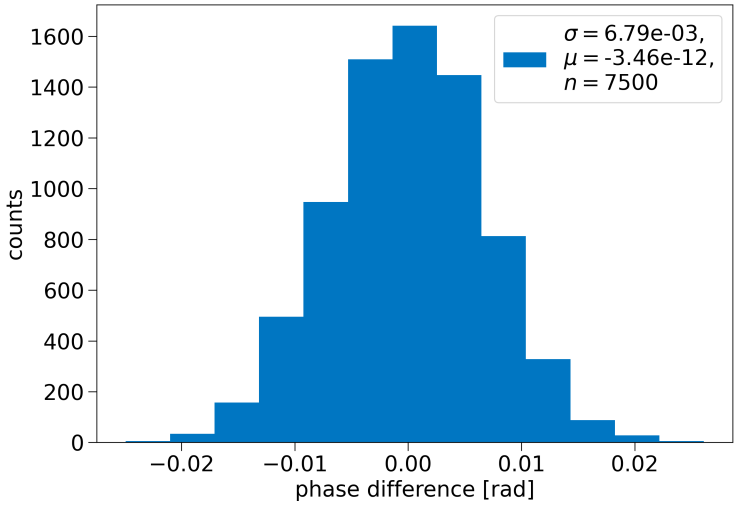


Figure 5.5: Histogram of the high-frequency phase fluctuations for 7500 phase measurements.

To increase the accuracy of the measurement of the phase resolution, multiple intervals are selected. The standard deviation of the phase resolution measurements is calculated for all of them. Intervals that had a resolution significantly different from the others were removed. To combine these multiple measurements, a weighted average σ_{avg} of the standard deviation is taken (Cohen, 1988)

$$\sigma_{avg} = \sqrt{\frac{\sum_{i=1}^{k} (n_i - 1)\, \sigma_i^2}{\sum_{i=1}^{k} n_i - k}}, \tag{5.1}$$

with k the number of intervals for which the standard deviation has been calculated and σ_i and n_i the standard deviation and sample size of the interval i. The result of this example then equals $\sigma_{avg} = 7.1 \pm 0.4$ mrad. This is an average of nine linear segments with a total of approximately 48000 phase measurements.

5.2.2. Power-dependent measurements

The measured beat signal at the photodiode is a combination of the test and reference beam which recombine on the last beam splitter in the set-up. It is expected that the phase resolution depends on the power of the test and reference beam. Van der Schaaf (2020) states that the rms noise for one image point scales with the test power P_{test} and reference power P_{ref} for an image point as

$$\Delta\phi = \frac{\Delta A_{test}}{A_{test}} \approx \frac{\sqrt{n_{electric}^2 + n_{shot}^2}}{2 \cdot \sqrt{P_{ref} P_{test}} \cdot \sqrt{2T}}, \tag{5.2}$$

with $T = 33$ μs the data acquisition time for a pixel, $n_{electric}$ the electric noise level and n_{shot} the shot noise level, which is proportional to the total power on the photodiode ($n_{shot} \propto P_{tot} = P_{ref} + P_{test}$).

To characterize the prototype set-up, the dependence of the phase resolution on the test and reference beam power is determined. First, the test and reference powers are measured by blocking the beams one by one and measuring the amplitude of the DC signal of the photodiode box. The power P is calculated by dividing the DC amplitude V_{DC} by the transimpedance of the amplifier Z_{trans} in the photodiode box and by the responsivity of the photodiode R_λ

$$P = \frac{V_{DC}}{Z_{trans} R_\lambda}. \tag{5.3}$$

The transimpedance of the amplifier in the photodiode box equals $Z_{trans} = 20$ kΩ and the responsivity of photodiode equals $R_\lambda \approx 0.9$ A/W for a wavelength of 1064 nm (OSI Optoelectronics, n.d.). The measured DC amplitudes in the prototype set-up are $V_{DC,test} \approx 1.265$ V and $V_{DC,ref} \approx 0.595$ V, corresponding to a test and reference power of $P_{test} \approx 70$ μW and $P_{ref} \approx 33$ μW.

The calculated powers were cross-checked by measuring the total beam power with a power meter. The measured total beam powers are $P_{test,tot} = 8.76$ mW and $P_{ref,tot} = 6.52$ mW. The beams are assumed to be Gaussian distributed and to be centered on the photodiode. The laser power through a circular aperture of the size of the photodiode is then $P_{test} = 52.8$ μW and $P_{ref} = 39.3$ μW. This is similar to the powers calculated from the DC amplitudes. Therefore, the DC amplitudes will be used in this study to measure the beam power.

The measured phase resolution of the carrier signal in the prototype set-up with $P_{test} \approx 70$ μW and $P_{ref} \approx 33$ μW is $\Delta\phi = 7.1 \pm 0.4$ mrad. The example measurement in figure 5.5 is one of the measurements used to get this averaged resolution. The resolution between several samples differs since the linear segments are selected manually. Therefore, the usage of a linear fit is not completely accurate. This could increase the measured resolution.

To check the phase resolution of equation 5.2 over a wider range of powers, the beams need to be reduced in power. This is done with neutral density (ND) filters. Eight measurements of the phase resolution of the carrier signal have been conducted for different values of P_{ref}, the measured powers varied between $2.6 \cdot 10^{-9}$ W and $3.3 \cdot 10^{-5}$ W. The spacing of these measurement points is based on the available ND filters and therefore the step size is not regular. During these measurements, the test power was fixed at $P_{test} = 72.7$ μW. The phase resolution is calculated in the same manner as described above. Figure 5.6 shows the results of these measurements as blue dots. The red line is fitted with a least squared optimization through the data points based on the expected relation between the phase resolution and the test and reference power (equation 5.2)

$$\Delta\phi = \frac{a}{\sqrt{P_{ref} P_{test}}} + b. \tag{5.4}$$

The function is fitted to return the parameters a and b. This fit neglects that the shot noise depends on the total power. Different fit functions with the shot noise depending on the total power have been tested but did not converge. Parameter b is added as an offset on top of the predicted relation which made this fit converge. It can represent possible other effects that influence the phase resolution.

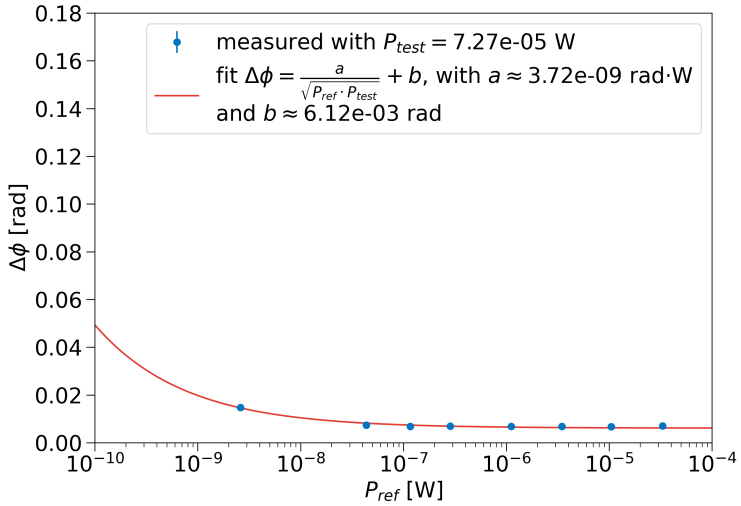


Figure 5.6: Phase resolution as a function of the reference power, P_{test} is fixed at 72.7 μW. The theoretical relation is fitted through the data points and shown as the red line.

The fit through the measured phase resolution for varying P_{ref} and $P_{test} = 72.7$ μW results in a value of $a \approx 3.72 \cdot 10^{-9}$ rad·W and $b \approx 6.12 \cdot 10^{-3}$ rad. This implies that there is a limit on the phase resolution which is not expected in equation 5.2.

A similar measurement was done to find the relation between phase resolution and varying P_{test}. Again a set of ND filters is used to reduce the power in the test beam. Besides, the reference beam was also reduced in power by a ND filter with an optical density of one such that $P_{ref} = 34.6$ μW. The test power

P_{test} was varied and measured between $2.2 \cdot 10^{-10}$ W and $7.4 \cdot 10^{-5}$ W. Figure 5.7 shows the measured phase resolution as blue points as well as the fitted function (equation 5.4) in red. The fit parameters are $a \approx 3.16 \cdot 10^{-9}$ rad·W and $b \approx 5.88 \cdot 10^{-3}$ rad, which are of the same order of magnitude as for the previous fit. The second fit is based on more data points over a wide range of powers than the first one and has a larger coefficient of determination, and therefore is more accurate.

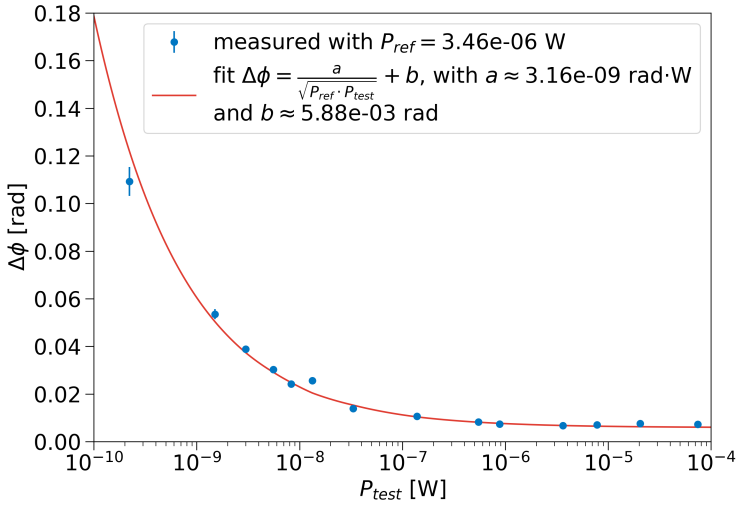


Figure 5.7: Phase resolution as a function of the test power, P_{ref} is fixed at 34.6 μW. The theoretical relation is fitted through the data points and shown as the red line.

These results can be compared with the expected rms noise on the phase measurements (equation 5.2). Figure 5.8 shows the expected phase resolution for a phase camera set-up for a range of test and reference powers. This image is adapted from Van der Schaaf (2020). The previous two measurements can be represented by a horizontal/vertical line in this image. From this image, the expected values a_{exp} and b_{exp} are estimated. The low bottom point and the red marked optimal point are used to calculate the two variables, $a_{exp} \approx 3 \cdot 10^{-10}$ and $b_{exp} \approx 0$.

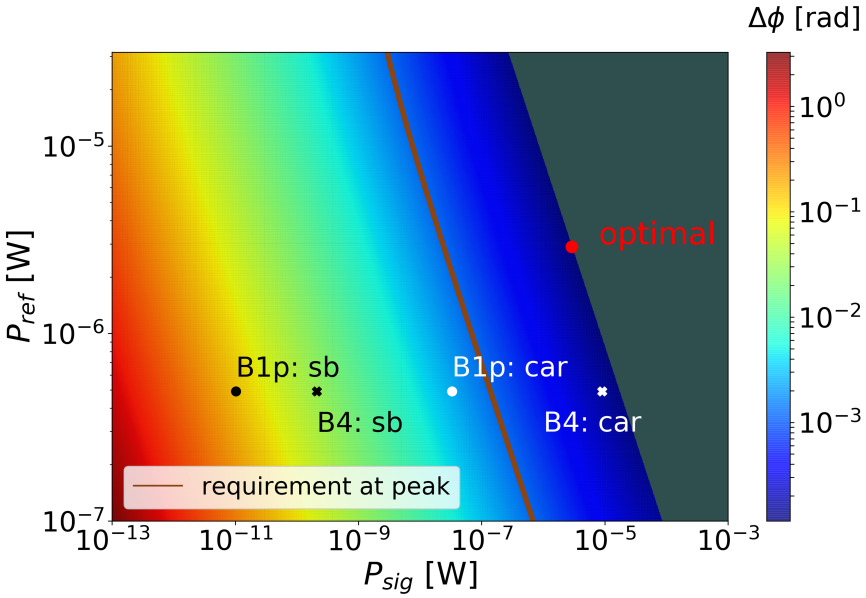


Figure 5.8: Expected resolution of the phase measurement for the phase camera versus a given test (signal) and reference power on the photodiode. The marker and labels in this plot are not relevant to this thesis. Adapted from Van der Schaaf (2020).

In table 5.1 the expected and fitted values for a and b are shown. Note that the fitted a is approximately one order of magnitude larger than expected and b is approximately 6 mrad larger. The difference in a could originate from uncertainties in the measured powers. The DC amplitudes of the measured signal of the photodiode fluctuate since the output power of the AOM is not stable. Van der Schaaf (2020) uses the same photodiode box as this research, therefore its electric noise component should be the same and thereby not the origin of the order of magnitude difference.

The offset b suggests that some other noise is present which creates an offset of approximately 6 mrad. Since this offset is power independent, it could be checked by connecting the output signal of the generators of the AOM and EOM directly to the ADC board which digitizes the signals for the FPGA. In this way, the effects of the optical set-up and the electric component of the photodiode box are removed. Besides possible electrical noise, this offset could be introduced by errors in the phase resolution data points due to the manual fitting of linear functions through segments of the data. Due to time limitations, a further noise study was not conducted. However, this is advised for the understanding of the power dependence of the phase resolution.

Table 5.1: Expected and fitted values of the parameters a and b from equation 5.4 to describe the relation between the phase noise and the test and reference power.

	a [rad·W]	b [rad]
expected	$\sim 3 \cdot 10^{-10}$	~ 0
fit varying P_{test}	$3.16 \cdot 10^{-9}$	$5.88 \cdot 10^{-3}$
fit varying P_{ref}	$3.72 \cdot 10^{-9}$	$6.12 \cdot 10^{-3}$

6

Discussion

In the previous chapters, a simulation of the phase images and the stability of the phase measurements are discussed. The results presented in these chapters are compared and discussed here. Possible improvements and further investigations are described. First, the centering of the phase image is discussed, followed by a discussion on the simulation of the phase images and the phase stability measurements.

6.1. Phase centering

Chapter 3 shows that the phase camera can measure both the intensity and the phase of a laser wavefront at different demodulated frequencies. In the current images, the centers of the intensity and phase images are not aligned. For example, in figure 3.9 the center of the phase images is approximately 0.5 mm more to the left and 0.2 mm more to the top than the center of the intensity image. This misalignment is caused by the optical set-up.

During this research, the set-up has been altered several times and the alignment of the images differed each time varying from tenths of millimeters difference up to a few millimeters. Due to time limitations, the images presented in this report are selected from the measurements where the centers were aligned up to tenths of milliliters, except for the measurements with varying photodiode positions. Those images are poorly aligned since the photodiode position strongly influences the alignment. Probably the misalignment is introduced by the test beam which reflects on the scanner mirror off-center. When the mirror rotates this introduces path length differences that are different than mathematically derived in chapter 4. Besides, the path length differences are not rotationally symmetrical around the center of the beam anymore. Therefore, the phase images are misaligned with respect to the intensity images. This alignment is difficult since the beam should be centered exactly onto the point of rotation, in the center, of the scanning mirror. This can be achieved by using two mirrors to steer the beam on the scanning mirror and using a beam profiler to measure the reflected beam. Now the mirror can be tilted and movement of the beam spot can be measured which should correspond to the tilt angle. To upgrade the prototype set-up this is the most important aspect.

6.2. Phase image simulation

The phase images are predicted in chapter 4. This mathematical derivation is used in the simulation of the phase images. In the comparison between the measured and simulated phase images (figure 4.7), some differences can be seen. The simulated number of rings in an image with a scan radius of 2 mm is 10 to 20% larger compared to measurements. Besides, the measured relation between the phase difference and photodiode position L_2 (figure 4.10) is similar to the simulation, but shifted. This hints that the parameters of the set-up that are used as input to the simulation are not fully correct. To quantify the influence of the input parameters a systematic study can be done by varying them one by one. Afterward, the dominant parameters should be measured again.

In the simulation, the phase difference is predominantly introduced by the path length difference of the AOM and EOM beam up to the beam splitter. Therefore, the placement of the components in the optical set-up should be checked for correspondence with the sketch in figure 4.3. For example, the laser beam should be perfectly horizontal along the entire beam path since a small inclination introduces extra optical path lengths compared to the mathematical derivation. Moreover, the beam splitter should be placed at exactly 45 degrees with respect to both beams to maximize the beam overlap.

6.3. Phase stability

The measurements in chapter 5 show that the measured phase fluctuates over time. Over time scales of tens of seconds, the phase fluctuations are probably dominated by temperature changes. Covering the set-up results in a significant increase since the airflow from the air-conditioning is blocked. Besides a cover, the set-up can be better isolated against temperature differences. Since an interferometer is a delicate apparatus, temperature is probably not the only origin of the fluctuations. The AOM is not completely stable. Its output power fluctuates on a scale of a few tenths of milliwatts. To check whether the phase of the output beam also fluctuates, the beam can be recombined on a beam splitter with a copy of the non-frequency shifted beam. The power behind the beam splitter can be measured and should oscillate constantly at 80 MHz when the phase is stable. Polarization fluctuations in the test and reference beam could also influence the measured phase. Interference only takes place when the polarization matches creating an extra term in equations 4.3 and 4.4 depending on the polarization overlap. Therefore, polarization fluctuations can be misinterpreted as phase fluctuations. To quantify polarization fluctuations, the power behind a polarizing beam splitter can be measured.

The phase resolution on the measurements improves with increasing test and reference beam powers. The expected relation (equation 5.2) is confirmed with two measurements, one for varying test powers and one for varying reference powers. The fits to these measurements show that the phase resolution is one order of magnitude larger than expected by Van der Schaaf (2020). This can be caused by errors in the measured powers which fluctuate. Besides, a constant offset to the phase resolution of approximately 6 mrad is found. This extra noise source could be electrical noise outside the photodiode box or errors in the measurement procedure of the phase resolution. A further noise study should be conducted to explain the differences between the measured and expected phase resolution.

7

Summary & conclusion

A phase camera set-up similar to the set-ups installed in Virgo has been built at Nikhef, with the aim to study its performance in a controlled environment. First measurements have been performed to check the operation of the camera. The prototype set-up has a one-beam scanning configuration. The studied heterodyne beat signal has a carrier frequency of 80 MHz and sidebands at 75 MHz and 85 MHz. Wavefronts at these frequencies were simultaneously acquired, digitally demodulated and phase maps were reconstructed.

For a better understanding of the phase images, a systematic study has been conducted. The interference of two Gaussian beams with a scanning mirror was mathematically derived and used in a simulation to predict phase images based on the layout of the optical set-up. The one-beam scanning configuration gives an additional phase that depends on the radial distance from the center. Phase wrapping introduces ring-like structures in the phase images. The predicted visible number of rings in the images is in the order of 10 to 20% larger compared to measurements. Besides, the derived dependence of the number of visible rings on the photodiode position is similar as measured. Therefore, the results of the simulation are encouraging.

Measurements of the phase stability show that the prototype set-up should be covered to prevent airflow from the air-conditioning. In the covered set-up the fluctuations are a factor four lower. However, low-frequency phase fluctuations are still a factor ten larger than high-frequency pixel-to-pixel fluctuations. The high-frequency phase resolution reduces with increasing test and reference power as physically expected. In the prototype set-up, the resolution is approximately $\Delta\phi = 7.1 \pm 0.4$ mrad for $P_{test} \approx 70\ \mu$W and $P_{ref} \approx 33\ \mu$W. Power-dependent measurements show that the phase resolution is one order of magnitude larger than expected by Van der Schaaf (2020).

The prototype camera can be improved by aligning the centers of the intensity and phase images to increase the resolution of the blurred edges of the phase images. Further improvements are a noise study of the set-up as well as the creation of a permanent cover to reduce thermal fluctuations. The phase image simulation can be optimized by accurately measuring the input parameters of the simulation. When the current set-up is fully tested, an aberration beam can be added to add controlled distortions to the images to understand the effect and origin of measured distortions.

All in all, the results of this research are promising. Further characterization of the prototype set-up will help the understanding of this type of phase camera and eventually will help understand higher-order modes, thereby improving the sensitivity of the Virgo gravitational wave detector.

Bibliography

Abbott, B. P., Abbott, R., Abbott, T. D., Abernathy, M. R., Acernese, F., Ackley, K., Adams, C., Adams, T., Addesso, P., Adhikari, R. X., Adya, V. B., Affeldt, C., Agathos, M., Agatsuma, K., Aggarwal, N., Aguiar, O. D., Aiello, L., Ain, A., Ajith, P., … Zweizig, J. (2016). Observation of Gravitational Waves from a Binary Black Hole Merger. *Physical Review Letters*, *116*, 061102.

Acernese, F., Agathos, M., Agatsuma, K., Aisa, D., Allemandou, N., Allocca, A., Amarni, J., Astone, P., Balestri, G., Ballardin, G., Barone, F., Baronick, J.-P., Barsuglia, M., Basti, A., Basti, F., Bauer, T. S., Bavigadda, V., Bejger, M., Beker, M. G., … Zendri, J.-P. (2014). Advanced Virgo: a second-generation interferometric gravitational wave detector. *Classical and Quantum Gravity*, *32*(2), 024001.

Agatsuma, K., Van der Schaaf, L., Van Beuzekom, M., Rabeling, D., & Van den Brand, J. (2019). High-performance phase camera as a frequency selective laser wavefront sensor for gravitational wave detectors. *Optics Express*, *27*(13), 18533–18548.

Bond, C., Brown, D., Freise, A., & Strain, K. A. (2016). Interferometer techniques for gravitational-wave detection. *Living Reviews in Relativity*, *19*(3), 1–217.

Cervantes-Cota, J., Galindo-Uribarri, S., & Smoot, G. (2016). A Brief History of Gravitational Waves. *Universe*, *2*(3), 22.

CINOGY Technologies. (2013). RayCi: Laser Beam Profiling Software [version 2.3.4, standard edition]. `http://www.cinogy.com/html/downloads___.html`

Cohen, J. (1988). *Statistical Power Analysis for the Behavioral Sciences* (2nd ed.). Lawrence Erlbaum Associates, Publishers.

CVI Melles Griot. (n.d.). Gaussian Beam Optics [Historic CVI Melles Griot Technical Guide]. IDEX Health & Science. `https://www.idex-hs.com/wp-content/uploads/2020/10/CVIMellesGriotTechGuide.zip`

Day, R. (2013). Using the phase camera in advanced interferometers [Presented in the Gravitational-waves Advanced Detector Workshop (VIR-0310A-13) in Elba, Italy, on the 19th till the 25th of May 2013].

Einstein, A. (1915a). Grundgedanken der allgemeinen Relativitätstheorie und Anwendung dieser Theorie in der Astronomie. *Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften Berlin (Mathematik Physik)*, *1*, 315.

Einstein, A. (1915b). Zur allgemeinen Relativitätstheorie. *Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften Berlin (Mathematik Physik)*, *2*, 778–786, 799–801.

Einstein, A. (1915c). Erklärung der Perihelbewegung des Merkur aus der allgemeinen Relativitätstheorie. *Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften Berlin (Mathematik Physik)*, *3*, 831–839.

Einstein, A. (1915d). Feldgleichungen der Gravitation. *Sitzungsberichte der Königlich Preussischen Akademie der Wissenschaften Berlin (Mathematik Physik)*, *4*, 844–847.

Franzen, A. (2012). ComponentLibrary: A vector graphics library for illustrations of optics experiments. `https://www.gwoptics.org/ComponentLibrary/`

Freise, A., & Strain, K. (2010). Interferometer Techniques for Gravitational-Wave Detection. *Living Reviews in Relativity*, *13*(1), 53–75.

Goda, K., Ottaway, D., Connelly, B., Adhikari, R., Mavalvala, N., & Gretarsson, A. (2004). Frequency-resolving spatiotemporal wave-front sensor. *Optics Letters*, *29*(13), 1452–1454.

Gretarsson, A. M., D'Ambrosio, E., Frolov, V., O'Reilly, B., & Fritschel, P. K. (2007). Effects of mode degeneracy in the LIGO Livingston Observatory recycling cavity. *Journal of the Optical Society of America B*, *24*(11), 2821–2828.

Kogelnik, H., & Li, T. (1966). Laser Beams and Resonators. *Applied Optics*, *5*(10), 1550–1567.

McCarron, D. J. (2007). A guide to acousto-optic modulators [Technical report]. `http://themccarrongroup.com/wp-content/uploads/2020/03/AOM-Guide.pdf`

Oppenheim, A. V., Willsky, A. S., & Hamid Nawab, S. (2013). *Signals and Systems: Pearson New International Edition* (2nd ed.). Pearson Education Limited.

OSI Optoelectronics. (n.d.). *Datasheet High Speed InGaAs Photodiodes* [FCI-InGaAs-55].
Physik Instrumente. (2003). *User Manual PZ 125E* [S-334 Tip/Tilt Mirror].
Physik Instrumente. (2009). *User Manual PZ 77E* [E-509 Position Servo-Control Module].
Physik Instrumente. (2013). *User Manual PZ 62E* [E-500/E-501 Series, Modular Piezo Controller].
Qubig. (2020). *Test Data Sheet PS2M-NIR* [S/N: M11550].
Rocchi, A., Coccia, E., Fafone, V., Malvezzi, V., Minenkov, Y., & Sperandio, L. (2012). Thermal effects and their compensation in Advanced Virgo. *Journal of Physics: Conference Series*, *363*(1), 012016.
Rudakova, V. (2018). How to calculate angular (phase) average [Online; accessed 20 February 2023]. https://vicrucann.github.io/tutorials/phase-average/
Saleh, B. E. A., & Teich, M. C. (2009). *Fundamentals of photonics* (3rd ed.). John Wiley & Sons, Inc.
Self, S. A. (1983). Focusing of spherical Gaussian beams. *Applied Optics*, *22*(5), 658–661.
Svelto, O., & Hanna, D. C. (2010). *Principles of lasers* (5th ed.). Springer New York, NY.
Thüring, A., & Lastzka, N. (2011). JamMt: Just another mode matching tool [version 0.24]. https://www.sr.bham.ac.uk/dokuwiki/doku.php?id=geosim:jammt#jammt_-_just_another_mode_matching_tool
Van Beuzekom, M., Tacca, M., Pascucci, D., Guo, Y., Van der Schaaf, L., & Van den Brand, J. (2019). Virgo Phase Camera [Presented at the Nikhef Jamboree in Amsterdam, the Netherlands, on the 16th and 17th of December 2019].
Van der Schaaf, L. (2020). *The Phase Cameras of Advanced Virgo* [Doctoral dissertation, Vrije Universiteit Amsterdam].
Weiss, R. (1972). Electronically Coupled Broadband Gravitational Antenna. *Quarterly Progress Report, Research Laboratory of Electronics, MIT*, *105*, 54.
Zangwill, A. (2012). *Modern Electrodynamics* (1st ed.). Cambridge University Press.
Zetie, K. P., Adams, S. F., & Tocknell, R. M. (2000). How does a Mach-Zehnder interferometer work? *Physics Education*, *35*(1), 46–48.

Appendices

A

Laguerre-Gaussian modes

The Laguerre-Gaussian (LG) modes are, in contrast to the Hermite-Gaussian modes (section 2.1.4.1), described in the polar coordinate system. This set of solutions is complete too. Therefore, it can be used to describe the electric field as a linear combination

$$\vec{E}\,(x,y,z) = \sum_{pl} A_{pl} U_{pl}\,(r,\phi,z)\, e^{-ikz+i\omega t}\vec{e}. \tag{A.1}$$

The two indices $p \in \mathbb{N}$ and $l \in \mathbb{Z}$ are the radial and azimuthal mode index, respectively. The normalization of the Laguerre-Gaussian modes is defined as

$$\int U_{pl}\,(r,\phi,z)\, U^{*}_{qo}\,(r,\phi,z)\, r dr d\phi = \begin{cases} 1 & \text{for } p = q \text{ and } l = o \\ 0 & \text{else} \end{cases}. \tag{A.2}$$

The Laguerre-Gaussian modes are given by (Van der Schaaf, 2020)

$$U_{pl}\,(r,\phi,z) = \sqrt{\frac{2p!}{\pi\,(p+|l|)!}}\,\frac{1}{w\,(z)}\left(\frac{\sqrt{2}r}{w\,(z)}\right)^{|l|} e^{-\frac{r^2}{w(z)^2}} L_p^{|l|}\left(\frac{2r^2}{w\,(z)^2}\right) e^{-ik\frac{r^2}{2R(z)}+il\phi+i(|l|+2p+1)\Phi(z)}, \tag{A.3}$$

where $L_p^{|l|}$ is a generalized Laguerre polynomial. The first few Laguerre polynomials are

$$L_0^{|l|}\,(x) = 1, \quad L_1^{|l|}\,(x) = 1 + |l| - x \quad \text{and} \quad H_3\,(x) = \tfrac{1}{2}\,(|l|+1)\,(|l|+2) - (|l|+2)\,x + \tfrac{1}{2}x^2. \tag{A.4}$$

The amplitude and phase of the Laguerre-Gaussian modes $U_{10}\,(r,\phi,z)$ and $U_{01}\,(r,\phi,z)$ are shown in figure A.1 for a beam with $\lambda = 1064$ nm, $w_0 = 1$ mm, $z_0 = 0$ mm at $z = 2$ m. Note, the amplitudes of the LG modes are always radially symmetric since they are independent of ϕ. For the phase images, this is only true for $l = 0$. Larger values of the azimuthal mode index create spirals in the phase images. Therefore, all LG modes $U_{p0}\,(r,\phi,z)$ are fully radially symmetric.

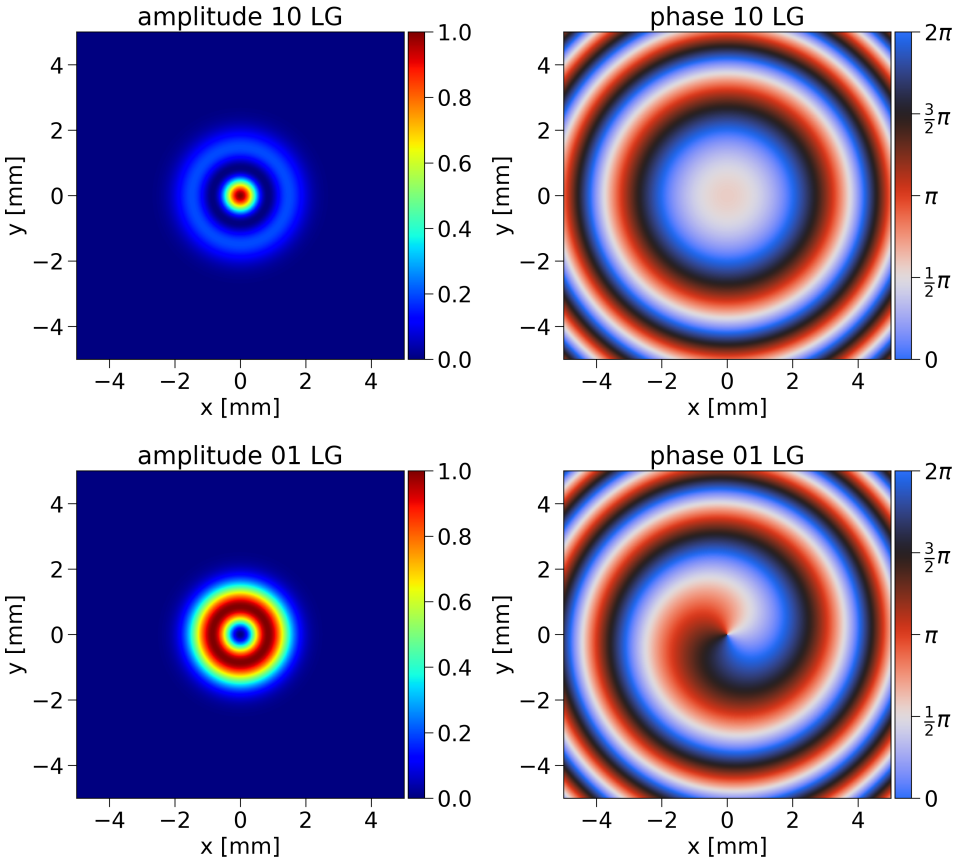


Figure A.1: Amplitude and phase images for the Laguerre-Gaussian modes $U_{10}(r, \phi, z)$ (upper row) and $U_{01}(r, \phi, z)$ (bottom row). The modes are for a beam with $\lambda = 1064$ nm, $w_0 = 1$ mm, $z_0 = 0$ mm at $z = 2$ m.

B

Python scripts

For this thesis, several Python scripts were written for simulations and data processing. The two most relevant scripts are attached here.

B.1. Data processing

In section 3.2.5, the demodulation of the measured signal is explained. Besides, the data processing is described which is used to create intensity and phase images from batched frame files. Below, the Python script that is used for this data processing is attached.

```
import sys
sys.path.append('../Python')

import pickle
import colorcet
import numpy as np
import parameters as par
import matplotlib.pyplot as plt
from astropy.time import Time
from matplotlib.ticker import AutoMinorLocator
from mpl_toolkits.axes_grid1 import make_axes_locatable

class phase_camera_frame:
    """
    Frame of measurement of the phase camera
    """

    # The init method or constructor
    def __init__(self, frametime, pickle_dict, gwf_filetime):
        # Instance Variable
        self.ndatapoints = 16384
        self.low_res_dict = {}

        self.gwf_filetime = gwf_filetime
        self.frametime = frametime

        # Instance Functions
        self.utc_time()
        self.load_channels(pickle_dict)

    def __getattr__(self, channel):
        # Try to read one channel of the frame
        try:
            return getattr(self, channel)
        except:
            # If channel is not in the data return error message
            print('Could not find channel %s in frame!' % channel)

    def utc_time(self):
```

```
        # Read GPS and calculate UTC time of frame
        frametime_gps = Time(self.frametime, format='gps')
        self.frametime_utc = Time(frametime_gps, format='iso', scale='utc')

    def load_channels(self, pickle_dict):
        print("   Load frame: "+str(self.frametime)+" GPS / "+str(self.frametime_utc)+ " UTC"
    )

        # Get all channel names with unique frequencies and create a list containing them
        self.channel_names = list(set([key.split("MHz")[0]+"MHz" for key in pickle_dict.keys
    () if "MHz" in key]))

        # Save all channels in frame
        for key in pickle_dict.keys():
            self.__setattr__(key, pickle_dict[key].flatten())

    def get_power(self):
        # For all channels calculate the power distribution in each bin
        for channel_name in self.channel_names:
            # Get the amplitude of the channel and square it to get power
            power = (self.__getattr__(channel_name + "_AMP"))**2

            # Create a histogram of the power and take the mean in each bin
            Power = self.accum2d(power)
            Power[self.valid] /= self.N[self.valid]

            # Bins without samples are set NaN for blank background in the image
            Power[self.N==0] = np.NaN

            # Check which pixels have power lower than the cut-off power
            low_res = Power < self.power_cut_off
            #low_res = Power < np.nanmax(Power)*np.exp(-2)
            self.low_res_dict[channel_name] = low_res

            # Save the power image in a channel
            self.__setattr__(channel_name + "_PWR", Power)

    def get_phase(self):
        # For all channels calculate the relative phase in each bin
        for channel_name in self.channel_names:
            # Get the phase and reference phase of the channel and take the difference
            phi = self.__getattr__(channel_name+"_PHI")
            phi -= self.__getattr__(channel_name+"_PHI_REF")

           # Calculate the real and imaginary component of the phase vector
            phi_x = np.cos(phi)
            phi_y = np.sin(phi)

            # Create a weighted histogram for the real and imaginary part of the phase vector
            Phi_x = self.accum2d(phi_x)
            Phi_y = self.accum2d(phi_y)

            # Take the argument of each bin of the histograms to get the "average" phase and
    map it to the 0 to 2*pi domain
            Phi = np.arctan2(Phi_y,Phi_x)
            Phi = np.fmod(Phi+2*np.pi,2*np.pi)

            # Bins without samples are set NaN for blank background in the image
            Phi[self.N==0] = np.NaN

            # The pixels with power lower than the cut-off power are set a NaN to remove them
            low_res = self.low_res_dict[channel_name]
            Phi[low_res] = np.NaN

            # Save the phase image in a channel
            self.__setattr__(channel_name + "_PHASE", Phi)

    def scale(self, x, x_min, x_max, n):
        """scale x from the interval x_min..x_max to an interger 0..n-1"""
        x_scaled = x - x_min
        x_scaled *= n / (x_max - x_min)
```

```
        np.floor(x_scaled, x_scaled)

        x_int = x_scaled.astype(int)
        np.clip(x_int, 0, n-1, out=x_int)
        return x_int

    def accum2d(self, weights):
        """create 2D (weighted) histogram, see bincount"""
        result = np.bincount(self.spiral_iflat , weights, self.im_shape[0] * self.im_shape
    [1])
        return result.reshape(self.im_shape)

    def get_spiral(self):
        # Define image properties (number of pixels in x and y direction)
        nx, ny = 100, 100
        self.im_shape = (nx, ny)

        # Some constants for scanning pattern
        nspiral = self.ndatapoints                    # number of samples in the spiral
    pattern
        tsample = nspiral * 2.0e-9                    # sampling time of FPGA
        t_fpga = np.arange(nspiral) * tsample         # time array for the FPGA
        t_pzt_Proto = np.arange(100000) / 100000      # time array scanner (piezo elements)
        PD_to_scanner_distance = self.dist_PD_scanner # distance from scanner to photodiode

        if self.scanning_pattern == "manual":
            # Some constant for the manual scanning pattern
            freqsample = 120                          # rotational frequency of the scanner
            ttotal = nspiral * tsample                # total time for one scanning pattern
            Timg = ttotal                             # total time for one image
            Rimg = 0.005                              # radius of the image

            # Manual creation of the x- and y-coordinates of the scanning pattern
            pattern_x = Rimg * t_pzt_Proto/Timg * np.cos(2*np.pi*freqsample*t_pzt_Proto)
            pattern_y = Rimg * t_pzt_Proto/Timg * np.sin(2*np.pi*freqsample*t_pzt_Proto)

        elif self.scanning_pattern == "measured":
            # Calculate x- and y-coordinates of the scanning pattern from the measured strain
     gauges
            # Take scanning pattern measured from scanner (calibration 24.04.2023)
            alpha1 = 0.001761004167562993
            alpha2 = -0.001630598507892687
            out_chan_1 = (self.__getattr__("ProtoPC_StrainGauge_axis2") * 57.16523963847714 -
     15.37745792378949) * alpha1
            out_chan_2 = (self.__getattr__("ProtoPC_StrainGauge_axis1") * 61.63159059032622 -
     15.20447582239593) * alpha2

            # Convert piezo channels to scanning pattern
            pattern_x = np.sqrt(2)*PD_to_scanner_distance*(out_chan_2-out_chan_1)
            pattern_y = -PD_to_scanner_distance*(out_chan_1+out_chan_2)

        else:
            # Print error message if no valid option for scanning_pattern is entered
            print("Error: '" + str(self.scanning_pattern) + "' is no valid option for
    scanning_pattern, choose: 'measured' or 'manual'")
            exit()

        # Interpolate to get the scanner positions at the photodiode PD measured times
        spiral_x = np.interp(t_fpga, t_pzt_Proto, pattern_x)
        spiral_y = np.interp(t_fpga, t_pzt_Proto, pattern_y)

        # Get the minimum and maximum values of the scanner positions in x- and y-direction
        x_min, x_max = spiral_x.min(), spiral_x.max()
        y_min, y_max = spiral_y.min(), spiral_y.max()

        # Convert spiral to a flat index into a 2D matrix, returns (n,m) positions in image
    matrix for each position
        ix = self.scale(spiral_x, x_min, x_max, self.im_shape[0])
        iy = self.scale(spiral_y, y_min, y_max, self.im_shape[1])

        # Converts (n,m) position in the image matrix into a pixel number
```

```
172         self.spiral_iflat = np.ravel_multi_index((iy, ix), (ny, nx))
173 
174         # Calculate number of samples per bin for normalization
175         self.N = self.accum2d(None)
176         self.valid = self.N > 0
177 
178         # Create X and Y mesh for plot
179         x = np.linspace(x_min, x_max, nx+1)
180         y = np.linspace(y_min, y_max, ny+1)
181         self.X, self.Y = np.meshgrid(x, y)
182 
183     def read_frame(self, setup_parameters):
184         print("   Read frame: "+str(self.frametime)+" GPS / "+str(self.frametime_utc)+ " UTC"
    )
185 
186         # Get initial parameters to read frame
187         self.scanning_pattern = setup_parameters["scanning_pattern"]
188         self.dist_PD_scanner = setup_parameters["dist_PD_scanner"]
189         self.power_cut_off = setup_parameters["power_cut_off"]
190 
191         # Create spiral and read power and phase
192         self.get_spiral()
193         self.get_power()
194         self.get_phase()
195 
196     def plot_frame(self, plot_parameters):
197         print("   Plot frame: "+str(self.frametime)+" GPS / "+str(self.frametime_utc)+ " UTC"
    )
198 
199         # Define which frequencies should be plotted
200         n_sidebands = plot_parameters["n_sidebands"]
201         freqs = [plot_parameters["AOM_freq"]+plot_parameters["EOM_freq"]*i for i in range(-
    n_sidebands,n_sidebands+1)]
202 
203         # Define colormap for amplitude images and set pixels without data to black
204         colormap_amp = plt.get_cmap(plot_parameters["colormap_ampl_image"]).copy()
205         colormap_amp.set_bad(color = par.tud_colors["black"])
206 
207         # Define colormap for amplitude images and set pixels without data to black
208         colormap_phase = plt.get_cmap(plot_parameters["colormap_phase_image"]).copy()
209         colormap_phase.set_bad(color = par.tud_colors["black"])
210 
211         # Define plot
212         fig, axs = plt.subplots(2,2*n_sidebands+1, figsize=(18,12))
213         plt.subplots_adjust(left=0.055, bottom=0.03, right=0.95, top=0.95, wspace=0.45,
    hspace=0.025)
214 
215         # Set plot title
216         fig.suptitle(str(self.frametime)+" GPS / "+str(self.frametime_utc)+ " UTC")
217 
218         # Loop over all frequencies to plot
219         for j in range(len(freqs)):
220             # Read plot parameters
221             channel_name = plot_parameters["channels_dict"][freqs[j]]
222 
223             # Set axes
224             axs[0,j].set(aspect="equal", title="%d MHz" %freqs[j], xlabel="x [mm]", ylabel="y
     [mm]")
225             axs[0,j].xaxis.set_minor_locator(AutoMinorLocator(4))
226             axs[0,j].yaxis.set_minor_locator(AutoMinorLocator(4))
227             axs[1,j].set(aspect="equal", title="%d MHz" %freqs[j], xlabel="x [mm]", ylabel="y
     [mm]")
228             axs[1,j].xaxis.set_minor_locator(AutoMinorLocator(4))
229             axs[1,j].yaxis.set_minor_locator(AutoMinorLocator(4))
230 
231             # Plot power and phase
232             im0 = axs[0,j].pcolormesh(self.X*1e3,self.Y*1e3,self.__getattr__(channel_name+"
    _PWR"), shading='flat', cmap=colormap_amp, vmin=0)
233             im1 = axs[1,j].pcolormesh(self.X*1e3,self.Y*1e3,self.__getattr__(channel_name+"
    _PHASE"), shading='flat', cmap=colormap_phase, vmin=0, vmax=2*np.pi)
234 
```

```
235             # Define location for coloraxis
236             cax0 = make_axes_locatable(axs[0,j]).append_axes("right", size="5%", pad=0.05)
237             cax1 = make_axes_locatable(axs[1,j]).append_axes("right", size="5%", pad=0.05)
238 
239             # Set color axes
240             cbar0 = fig.colorbar(im0, cax=cax0, label="intensity [a.u.]")
241             cbar0.formatter.set_powerlimits((0, 0))
242             cbar0.ax.yaxis.set_minor_locator(AutoMinorLocator(4))
243             cbar1 = fig.colorbar(im1, cax=cax1, label="relative phase [rad]", ticks=[0, np.pi
    /2, np.pi, 3/2*np.pi, 2*np.pi])
244             cbar1.ax.set_yticklabels([r'$0$', r'$\frac{1}{2}\pi$', r'$\pi$', r'$\frac{3}{2}\
    pi$', r'$2\pi$'])
245             cbar1.ax.yaxis.set_minor_locator(AutoMinorLocator(4))
246 
247         fig.tight_layout()
248         plt.draw()
249         # plt.show()
250         plt.savefig(plot_parameters["storage_directory_plots"]+str(self.frametime)+".png")
251         plt.close()
252 
253     def save_information(self, channel_name):
254         # Save channel information to csv file for debugging
255         np.savetxt('data/simulation_validation/'+str(self.frametime)+'_X.csv', self.X*1e3,
    delimiter=';')
256         np.savetxt('data/simulation_validation/'+str(self.frametime)+'_Y.csv', self.Y*1e3,
    delimiter=';')
257         np.savetxt('data/simulation_validation/'+str(self.frametime)+'_phase_'+str(
    channel_name)+'.csv', self.__getattr__(channel_name+"_PHASE"), delimiter=';')
258         np.savetxt('data/simulation_validation/'+str(self.frametime)+'_power_'+str(
    channel_name)+'.csv', self.__getattr__(channel_name+"_PWR"), delimiter=';')
259 
260 def read_pickle_file(filetime):
261     # Path and file time of the pickle file to read
262     read_directory_files = "data/gwf_data/"
263     filename = "channels_" + str(filetime) + "_DC_vs_Car.pkl"
264     print("\nRead file:", filename)
265 
266     # Open pickle file and read it
267     with open(read_directory_files+filename, "rb") as file_pickled:
268         file = pickle.load(file_pickled, encoding='latin1')
269 
270     return filename, file
271 
272 def load_frame(frames, read_filetime, n_frames=1):
273     # Read pickle file
274     filename, file = read_pickle_file(read_filetime)
275 
276     # Time of first frame and number of frames in pickle file
277     gpstime = file["gpstime"]
278     duration = file["duration"]
279 
280     # Return error if file contains fewer frames than wanted to load
281     if n_frames > duration:
282         print("Cannot read",n_frames,"frames in "+str(filename)+". This file only contains",
    duration,"frames.")
283         return
284 
285     # Load number of frames in a dictionary
286     for time in np.arange(n_frames)+gpstime:
287         frames[time] = phase_camera_frame(time, file[time], gpstime)
288 
289     return frames
290 
291 def load_frames(filetimes, n_frames_per_file):
292     # Load multiple frames one by one
293     frames = {}
294     for filetime in filetimes:
295         frames = load_frame(frames, filetime, n_frames_per_file)
296 
297     return frames
298 
```

```
299 def initial_parameters(filetimes, dist_PD_scanner, storage_directory_plots):
300     ###### GENERAL PARAMETERS
301     ### Set-up parameters
302     # Frequencies of the AOM en EOM
303     AOM_freq = 80
304     EOM_freq = 5
305 
306     # Names of the channels in the DAQ
307     channels_dict = {55 : "",
308                      60 : "",
309                      65 : "",
310                      70 : "",
311                      75 : "Proto_B0_PC_RAW_14MHz",
312                      80 : "Proto_B0_PC_RAW_81MHz",
313                      85 : "Proto_B0_PC_RAW_145MHz",
314                      90 : "",
315                      95 : "",
316                      100 : "",
317                      105 : ""
318     }
319 
320     ### Plot parameters
321     # Paramters for scanning pattern to use, power cut off and number of sidebands to plot
322     scanning_pattern = "measured"
323     power_cut_off = 1e2
324     n_sidebands = 1
325 
326     colormap_ampl_image = "jet"
327     colormap_phase_image = "cet_CET_C3s"
328 
329     ###### PROCESSING
330     # Bundle parameters
331     setup_parameters_dict = {}
332     for i in range(len(filetimes)):
333         setup_parameters = {"dist_PD_scanner": dist_PD_scanner[i],
334                             "scanning_pattern": scanning_pattern,
335                             "power_cut_off": power_cut_off
336         }
337 
338         setup_parameters_dict[filetimes[i]] = setup_parameters
339 
340     plot_parameters = {"AOM_freq": AOM_freq,
341                        "EOM_freq": EOM_freq,
342                        "channels_dict": channels_dict,
343                        "n_sidebands": n_sidebands,
344                        "storage_directory_plots": storage_directory_plots,
345                        "colormap_ampl_image": colormap_ampl_image,
346                        "colormap_phase_image": colormap_phase_image
347     }
348 
349     return setup_parameters_dict, plot_parameters
350 
351 
352 ###### INPUT
353 ### File information
354 # Define filetimes of pickle files and distance between scanner and photodiode for each file
355 filetimes = np.array([1367924290])
356 dist_PD_scanner = np.array([0.85])
357 # filetimes = np.array([1366457250, 1366457330, 1366457400, 1366457460, 1366457530,
        1366457590, 1366457650, 1366457710, 1366457770, 1366457830, 1366457920, 1366458000,
        1366458080])
358 # dist_PD_scanner = np.array([0.85, 0.8, 0.75, 0.7, 0.65, 0.6, 0.55, 0.5, 0.45, 0.4, 0.35,
        0.3, 0.85])
359 
360 # Storage directory for plots
361 storage_directory_plots = "plots/PC/gwf_plots/"
362 
363 # Set initial parameters dictionary
364 setup_parameters_dict, plot_parameters = initial_parameters(filetimes, dist_PD_scanner,
        storage_directory_plots)
365 
```

```
# Load n frames per file
n_frames_per_file = 1
frames = load_frames(filetimes, n_frames_per_file)

# Set counters for plotting
number_of_plots = len(frames.keys())
plot_number = 0

# Loop over all frames
for frame in frames.values():
    plot_number += 1
    print("\nCreate image: "+str(plot_number)+"/"+str(number_of_plots))

    # Read frames
    frame.read_frame(setup_parameters_dict[frame.gwf_filetime])

    # Plot frames
    frame.plot_frame(plot_parameters)

    # Save information for debugging
    # frame.save_phase_information("Proto_B0_PC_RAW_81MHz")
```

B.2. Phase image simulation

In chapter 4, the phase images are predicted with a simulation (see section 4.5). The Python script that can simulate these phase images based on the parameters of the optical set-up is presented below.

```
import sys
sys.path.append('../Python')

import colorcet
import numpy as np
import beam_functions as bf
import matplotlib.pyplot as plt
from matplotlib.ticker import AutoMinorLocator

def wavenumber(sideband, delta_frequency_EOM, delta_frequency_AOM):
    """
    Calculates the wave numbers of the different section from their frequency
    """

    # Parameters
    wavelength = 1064e-9
    speed_of_light = 299792458

    # Laser frequency
    frequency = speed_of_light/wavelength

    # Frequencies of test and reference beam and beat signal
    frequency_EOM = frequency + sideband*delta_frequency_EOM
    frequency_AOM = frequency + delta_frequency_AOM
    frequency_beat = frequency_AOM - frequency_EOM

    # Wavelengths of test and reference beam and beat signal
    wavelength_EOM = speed_of_light/frequency_EOM
    wavelength_AOM = speed_of_light/frequency_AOM
    wavelength_beat = speed_of_light/frequency_beat

    # Wave numbers of test and reference beam and beat signal
    k_EOM = 2*np.pi/wavelength_EOM
    k_AOM = 2*np.pi/wavelength_AOM
    k_beat = 2*np.pi/wavelength_beat

    return k_EOM, k_AOM, k_beat

def path_length_calculations(L2, alpha, c, dy):
    """
    Calculate optical path lengths in the set-up
```

```
    """

    # Define an empty dictionary
    path_lengths = {}

    # Calculate L_1, L_1' and their difference
    path_lengths["L1"] = 0
    path_lengths["L1_prime"] = dy*np.tan(np.pi/4-alpha)
    path_lengths["dL1"] = path_lengths["L1_prime"] - path_lengths["L1"]

    # Calculate L_2' and the difference with L_2
    path_lengths["L2_prime"] = np.sqrt(path_lengths["dL1"]**2+(L2-dy)**2)
    path_lengths["dL2"] = path_lengths["L2_prime"] - L2

    # Calculate x and x'
    path_lengths["x"] = (np.tan(2*alpha))/(np.tan(2*alpha)+1)*(1-c)*L2
    path_lengths["x_prime"] = path_lengths["x"]/np.cos(2*alpha)

    # Calculate L_3, L_3' and their difference
    path_lengths["L3"] = c*L2
    path_lengths["L3_prime"] = (path_lengths["L3"]-dy+path_lengths["x"])/np.cos(2*alpha)
    path_lengths["dL3"] = path_lengths["L3_prime"] - path_lengths["L3"]

    # Calculate L_4, L_4' and their difference
    path_lengths["L4"] = (1-c)*L2
    path_lengths["L4_prime"] = (path_lengths["L4"]-path_lengths["x"])/np.cos(2*alpha)
    path_lengths["dL4"] = path_lengths["L4_prime"] - path_lengths["L4"]

    return path_lengths

def curved_wavefront_phase(y,k,z,w0,M,z0):
    """
    Calculates second phase term of Gaussian beam: k * ( (x^2+y^2) / (2*R(z)) )
    """

    # Calculate radius of curvature at position z
    R = bf.radius_of_curvature(z,w0,M,z0)

    # Calculate difference in z direction between point (x,y,z) and wavefront of z at (x,y,z
    ')
    dz = y**2/(2*R)

    # Multiply with wavenumber to calculate the phase difference
    phi_cw = k*dz

    return phi_cw

def path_length_difference_phase(k, dz):
    """
    Calculate phase difference due to path length difference
    """
    dphi_path_length = k*dz

    return dphi_path_length

def polar_to_cartesian(x, y, r_grid, theta_grid, data):
    """
    Map values from polar coordinate grid to cartesian coordinate grid
    """

    # Create empty array
    new = np.zeros_like(data) * np.nan

    # Loop over all elements of the coordinate grid
    for i in range(new.shape[0]):
        for j in range(new.shape[1]):
            # Take x and y coordinates and calculate the equivalent polar coordinates
            x0, y0 = x[j], y[i]
            r, a = np.sqrt(x0**2 + y0**2), np.arctan2(y0, x0)

            # Find closest matching coordinate on polar coordinate grid
```

```
            data_i = np.argmin(np.abs(theta_grid[:, 0] - a))
            data_j = np.argmin(np.abs(r_grid[0, :] - r))

            # Take value from polar coordinate grid
            val = data[data_i, data_j]

            # If radius within the image, then save value on cartesian grid
            if r <= np.max(x):
                new[i, j] = val

    return new

def plot_dL_vs_dy(dy, path_lengths, c, L2):
    """
    Plot path length difference upto the beam splitter
    """

    # Define figure and set title
    fig1 = plt.figure()
    plt.title("c = %.2f, L2 = %.2e m" %(c, L2))

    # Plot path length differences and total effect
    plt.plot(dy, path_lengths["dL1"], color="blue", label=r"$\Delta L_1$")
    plt.plot(dy, path_lengths["dL3"], color="red", label=r"$\Delta L_3$")
    plt.plot(dy, path_lengths["dL1"]+path_lengths["dL3"], color="black", label=r"$\Delta L_{
    EOM}$")

    # Define axes and create legend
    plt.xlabel(r"$\Delta y [m]$")
    plt.ylabel(r"$\Delta L_1$, $\Delta L_3$ and $\Delta L_{EOM}$ [m]")
    plt.ticklabel_format(style='sci', axis='both', scilimits=(0,0))
    plt.legend()

    # Save plot
    fig1.tight_layout()
    plt.savefig("plots/scanner_interference/dL_vs_dy.png")
    # plt.show()
    plt.close()

def plot_dphi_vs_dy(dy,dPhi,dlambda,c,L2, dphi_pl, dphi_cw):
    """
    Plot radial phase difference versus radial coordinate
    """

    # Define figure
    fig2 = plt.figure(figsize=(10,7))
    ax21 = fig2.add_subplot()

    # Plot curved wavefront effect, path length difference effect and total effect
    ax21.plot(dy*1e3, dPhi, color="black", label=r"$\Delta \phi_{total}$", linewidth=3)
    ax21.plot(dy*1e3, dphi_pl, color="blue", label=r"$\Delta \phi_{path~length}$", linewidth
    =2)
    ax21.plot(dy*1e3, dphi_cw, color="red", label=r"$\Delta \phi_{curved~wavefront}$",
    linewidth=2)

    # Create a second y-axis for the number of rings axis
    ax22 = ax21.twinx()
    ax22.plot(dy*1e3, dlambda, color="black")
    ymin, ymax = ax21.get_ylim()

    # Create title, define axes and create legend
    ax21.set_title("c = %.2f, L2 = %.2e m" %(c, L2))
    ax22.set_ylim([ymin/(2*np.pi), ymax/(2*np.pi)])
    ax21.set_xlabel(r"$\Delta y$ [mm]")#, fontsize=27)
    ax21.set_ylabel(r"$\Delta \phi$ [rad]")#, fontsize=27)
    ax22.set_ylabel(r"$\Delta L/\lambda = {\Delta \phi}/{2\pi}$")#, fontsize=27)
    ax21.ticklabel_format(style='sci', axis='x', scilimits=(0,0))
    ax21.legend()

    # Save plot
    fig2.tight_layout()
```

```
    plt.savefig("plots/scanner_interference/dphi_vs_dy.png")
    # plt.show()
    plt.close()

def plot_wavefront(dy, dPhi, dlambda, c, L2, dPhi_custom_offset):
    """
    Plot phase of the simulated wavefront
    """

    # Calculate polar coordinate system and create mesh
    angular_coord = np.radians(np.linspace(0,360,len(dy)))
    radial_coord = dy*1e3
    r,theta=np.meshgrid(radial_coord, angular_coord)

    # Repeat wavefront positions such that the line of values is rotated to get a circular
    wavefront
    dPhi += dPhi_custom_offset
    dPhi %= 2*np.pi
    wavefront = np.repeat(dPhi[np.newaxis,:],len(angular_coord), axis=0)

    # Define color map and color of empty pixels
    map=plt.get_cmap('cet_CET_C3s').copy()
    map.set_bad(color="black")

    # Define x and y coordinates and map wavefront to cartesian grid
    x = np.linspace(-np.max(radial_coord),np.max(radial_coord),len(radial_coord))
    y = np.linspace(-np.max(radial_coord),np.max(radial_coord),len(radial_coord))
    new_wavefront = polar_to_cartesian(x,y,r,theta,wavefront)

    # Define plot and set title
    fig3, ax3 = plt.subplots(figsize=(10,8))
    ax3.set(aspect="equal", title=r"$c =$%.2f, $L_2 =$%.2e m, $\Delta L=$%.2f$\lambda$" %(c,
    L2, dlambda[-1]))

    # Define axes
    plt.xlabel("x [mm]", fontsize=27)
    plt.ylabel("y [mm]", fontsize=27)
    ax3.xaxis.set_tick_params(labelsize=24)
    ax3.yaxis.set_tick_params(labelsize=24)
    ax3.xaxis.set_minor_locator(AutoMinorLocator(4))
    ax3.yaxis.set_minor_locator(AutoMinorLocator(4))

    # Plot wavefront
    cax=ax3.pcolormesh(x,y,new_wavefront, cmap=map, vmin=0, vmax=2*np.pi)

    # Define color axis
    cbar = fig3.colorbar(cax, ticks=[0, np.pi/2, np.pi, 3/2*np.pi, 2*np.pi])
    cbar.set_label("relative phase [rad]", size=27)
    cbar.ax.set_yticklabels([r'$0$', r'$\frac{1}{2}\pi$', r'$\pi$', r'$\frac{3}{2}\pi$', r'$2
    \pi$'], fontsize=24)
    cbar.ax.yaxis.set_minor_locator(AutoMinorLocator(4))

    # Save plot
    fig3.tight_layout()
    plt.savefig("plots/scanner_interference/dy_wavefront.png")
    # plt.show()
    plt.close()

def plot_ring_widths(dy, dlambda, c, L2):
    """
    Plot the number of rings versus the ring number
    """

    # Create array with the ring numbers
    lambda_int = np.arange(0,np.max(np.absolute(dlambda)))

    # Interpolate to get the coordinates of the edges of the rings
    ring_positions = np.interp(lambda_int, np.absolute(dlambda), dy)

    # Take the difference of the edges to get ring widths
    ring_widths = np.diff(ring_positions)
```

```
248
249     # Define figure and set title
250     fig4 = plt.figure(figsize=(10,7))
251     plt.title(r"$c =$%.2f, $L_2 =$%.2e m, $\Delta L=$%.2f$\lambda$" %(c, L2, dlambda[-1]))
252
253     # Plot ring width, set axis labels and draw gridlines
254     plt.plot(lambda_int[1:],ring_widths*1e3, color="blue", marker="o", linestyle="None",
        markersize=10)
255     plt.xlabel("ring number")
256     plt.ylabel("ring width [mm]")
257     plt.grid(True)
258
259     # Save plot
260     fig4.tight_layout()
261     plt.savefig("plots/scanner_interference/dy_ring_widths.png")
262     # plt.show()
263     plt.close()
264
265 ###### PARAMETERS
266 ### Set-up parameters
267 # Define scanning radius, L_2 and c
268 dy_max = 2e-3
269 L2 = 0.85
270 c = .225/L2
271
272 # Custom phase offset to the simulated wavefront
273 dPhi_custom_offset = 0
274
275 # Define EOM and AOM frequencies and which signal to study
276 delta_frequency_EOM = 5e6 #Hz
277 delta_frequency_AOM = 80e6 #Hz
278 sideband = 0 #integer from domain -5 to 5, 0 is carrier signal
279
280 ###### PROCESSING
281 # Calcuate wave numbers
282 k_EOM, k_AOM, k_beat = wavenumber(sideband, delta_frequency_EOM, delta_frequency_AOM)
283
284 # Define sampling of scanning radius
285 n_dy = 501
286 dy = np.linspace(0,dy_max,n_dy)
287
288 # Calculate scanning angles
289 alpha = 1/2 * np.arcsin(dy/L2)
290
291 # Calculate optical path lengths
292 path_lengths = path_length_calculations(L2, alpha, c, dy)
293
294 ### EOM beam
295 # Calculate path length to beam splitter for scanner in its stationary position
296 dz_EOM = path_lengths["dL1"] + path_lengths["dL3"]
297
298 # Beam radius at beam splitter
299 dy_EOM = dy
300
301 # Measured beam properties
302 w0_EOM = 8.02e-4
303 M_EOM = 1.03
304 z0_EOM = -0.352
305
306 # Calculate both phase effects
307 dphi_EOM_cw = curved_wavefront_phase(dy_EOM,k_EOM,dz_EOM,w0_EOM,M_EOM,z0_EOM)
308 dphi_EOM_pl = path_length_difference_phase(k_EOM, dz_EOM)
309
310 # Total phase effect
311 dphi_EOM = dphi_EOM_cw + dphi_EOM_pl
312
313 ### AOM beam
314 # Calculate path length to beam splitter
315 dz_AOM = path_lengths["x"]
316
317 # Beam radius at beam splitter
```

```
dy_AOM = path_lengths["x"]

# Measured beam properties
w0_AOM = 1.06e-3
M_AOM = 1.00
z0_AOM = 0.741

# Calculate both phase effects
dphi_AOM_cw = curved_wavefront_phase(dy_AOM,k_AOM,dz_AOM,w0_AOM,M_AOM,z0_AOM)
dphi_AOM_pl = path_length_difference_phase(k_AOM, dz_AOM)

# Total phase effect
dphi_AOM = dphi_AOM_cw + dphi_AOM_pl

### After beam splitter
# Calculate path length to photodiode
dz_beat = path_lengths["dL4"]

# Calculate phase effect due to path length difference
dphi_beat_pl = path_length_difference_phase(k_beat, dz_beat)

# Total phase effect
dphi_beat = dphi_beat_pl

### Phase effects
# Calculate both phase effects separately
dphi_cw = dphi_EOM_cw - dphi_AOM_cw
dphi_pl = dphi_EOM_pl - dphi_AOM_pl + dphi_beat_pl

# Total phase effect the whole configuration and equivalent number of rings
dPhi = dphi_EOM - dphi_AOM + dphi_beat
dlambda = dPhi/(2*np.pi)

###### PLOTTING
# Plot phase difference versus scanning radius
plot_dphi_vs_dy(dy,dPhi,dlambda,c,L2, dphi_pl, dphi_cw)

# Plot phase of the simulated wavefront
plot_wavefront(dy, dPhi, dlambda, c, L2, dPhi_custom_offset)

# Plot the ring widths
plot_ring_widths(dy, dlambda, c, L2)

# Plot path length differences
plot_dL_vs_dy(dy, path_lengths, c, L2)
```

C

Simulation validation measurements

Phase measurements to determine the relation between the number of phase rings and the photodiode position L_2. The figures are ordered in decreasing distance L_2. The results of this analysis are presented in section 4.6.3.

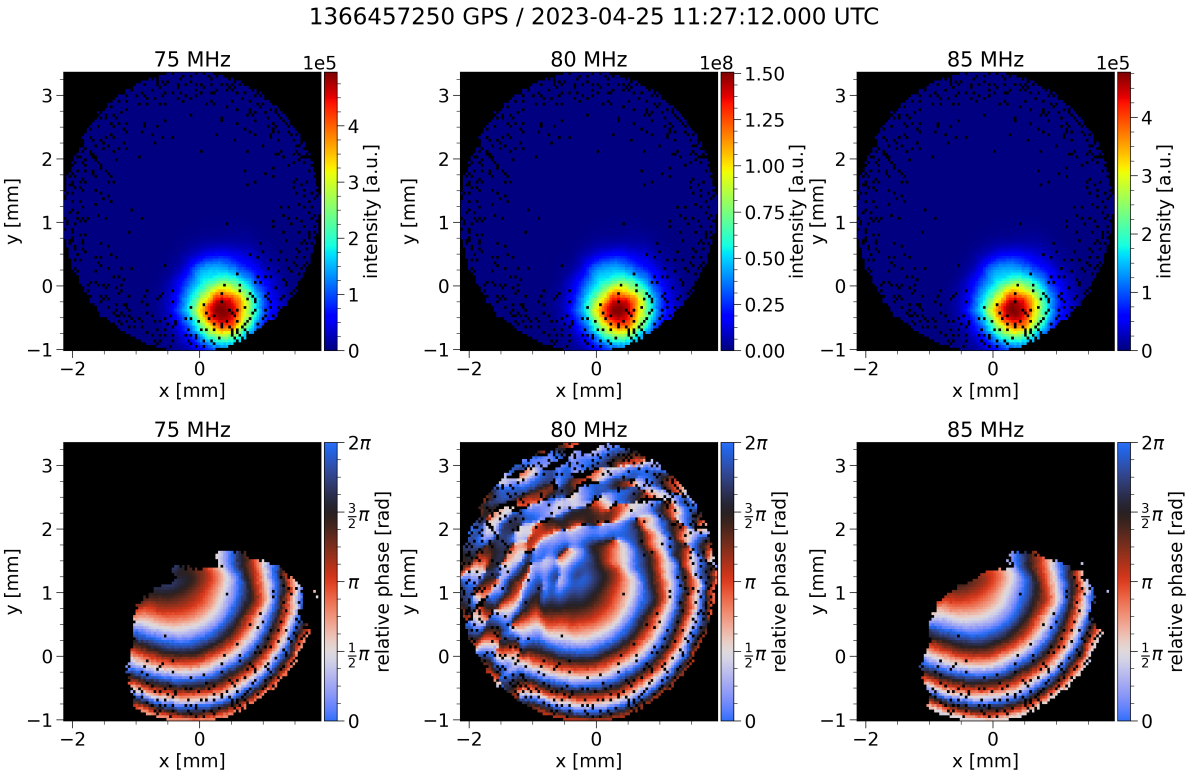


Figure C.1: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.85$ cm. The phase images are centered to measure the phase difference along the scanning radius.

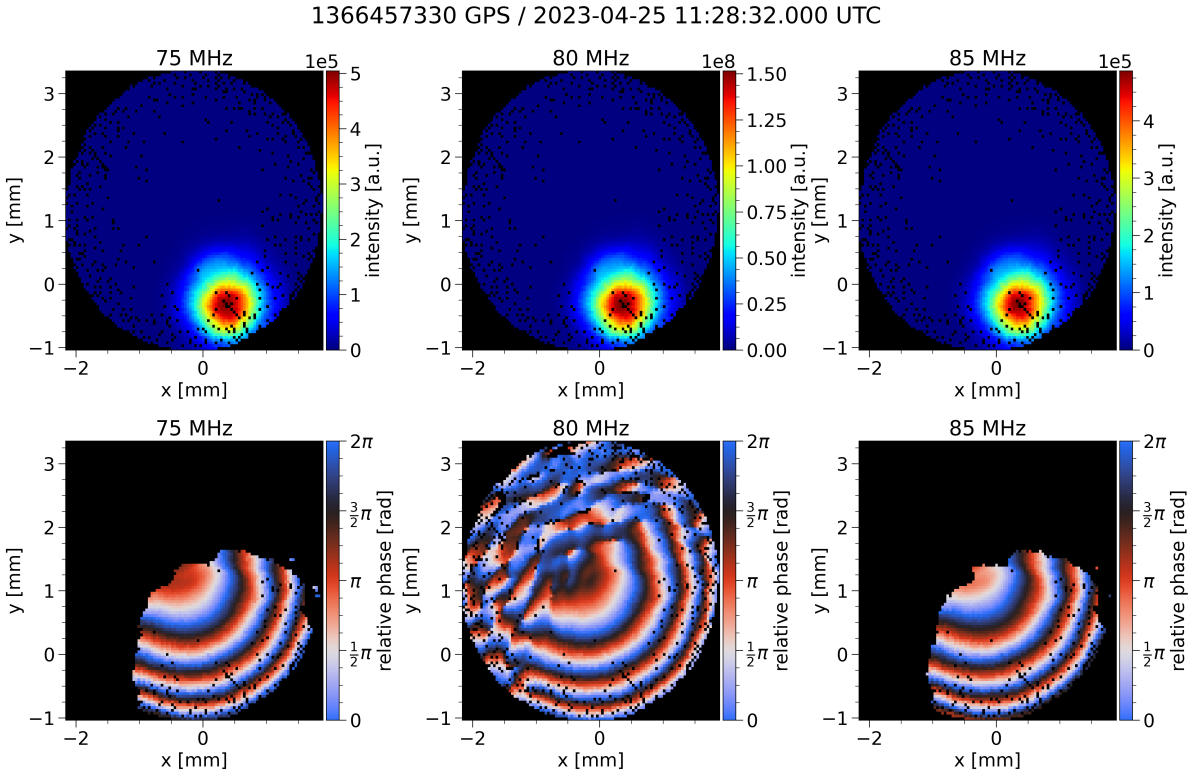


Figure C.2: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.80$ cm. The phase images are centered to measure the phase difference along the scanning radius.

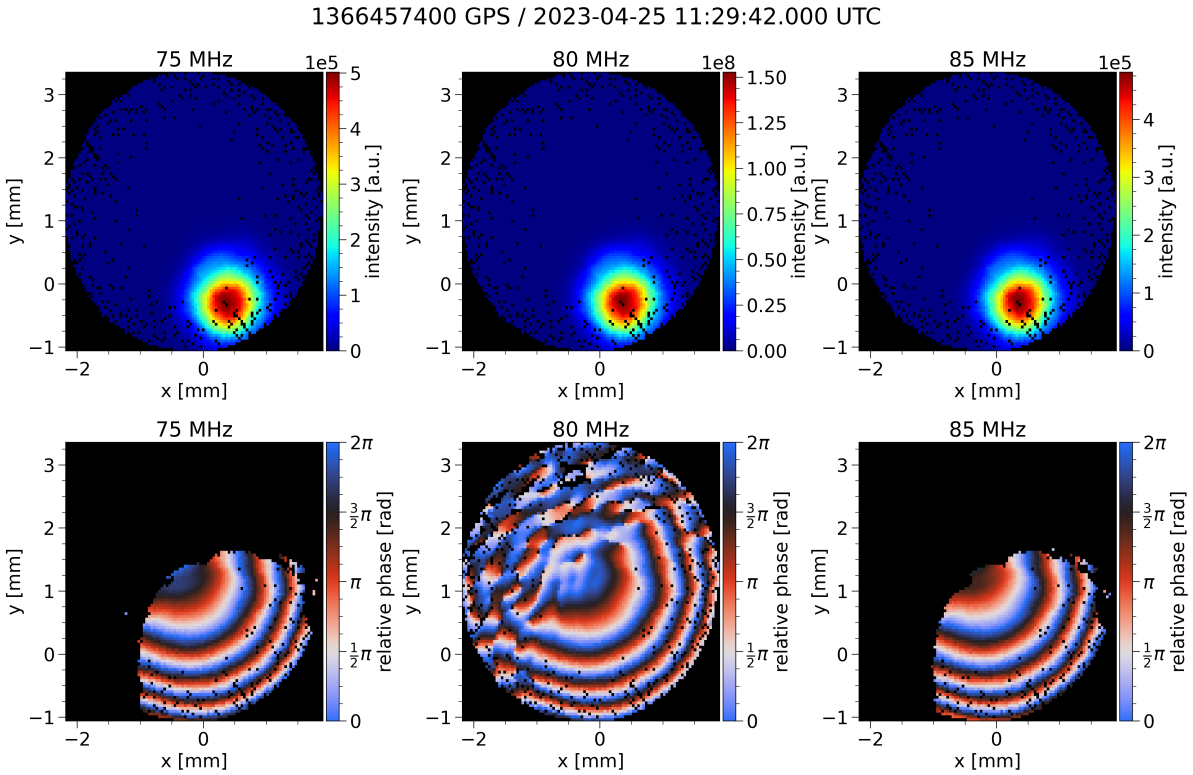


Figure C.3: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.75$ cm. The phase images are centered to measure the phase difference along the scanning radius.

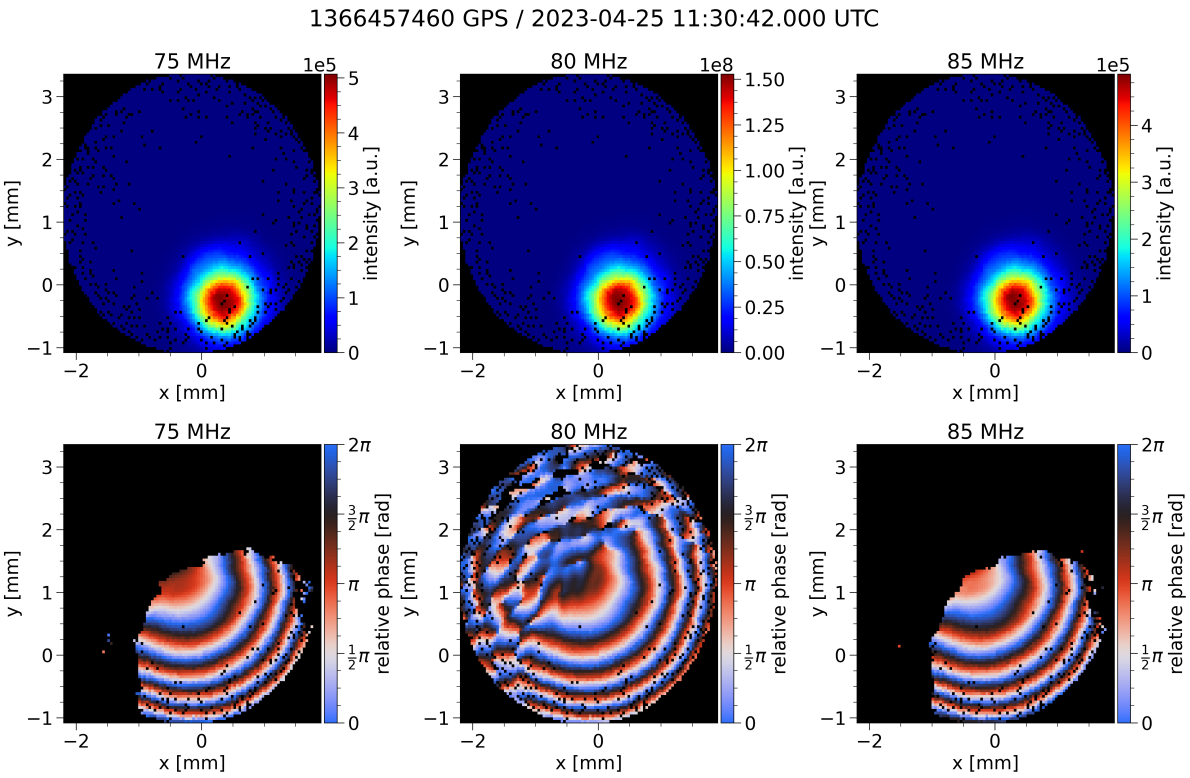


Figure C.4: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.70$ cm. The phase images are centered to measure the phase difference along the scanning radius.

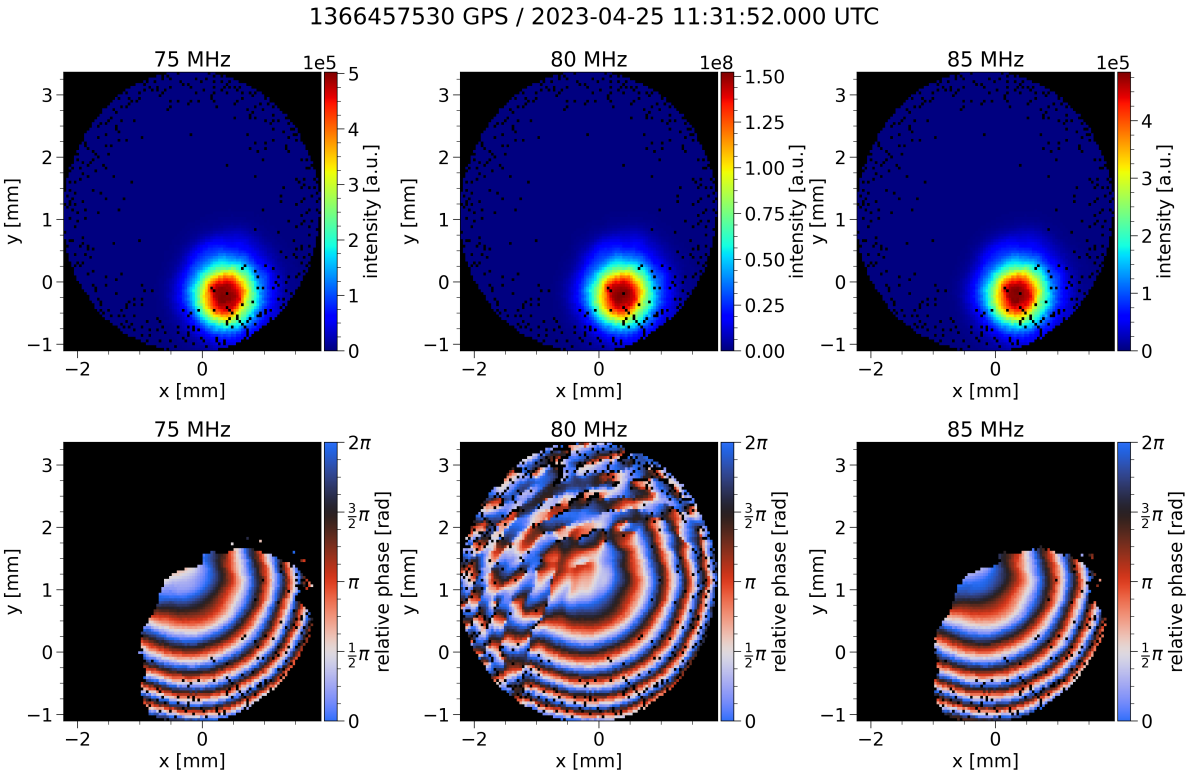


Figure C.5: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.65$ cm. The phase images are centered to measure the phase difference along the scanning radius.

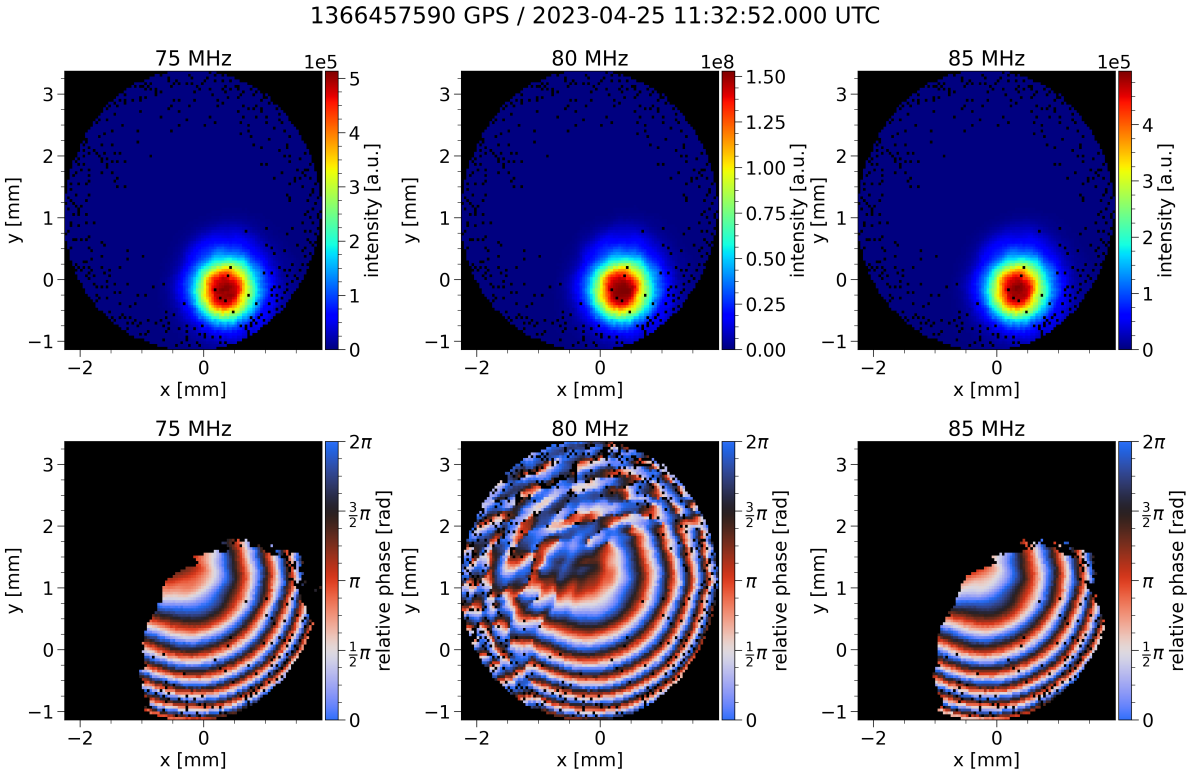


Figure C.6: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.60$ cm. The phase images are centered to measure the phase difference along the scanning radius.

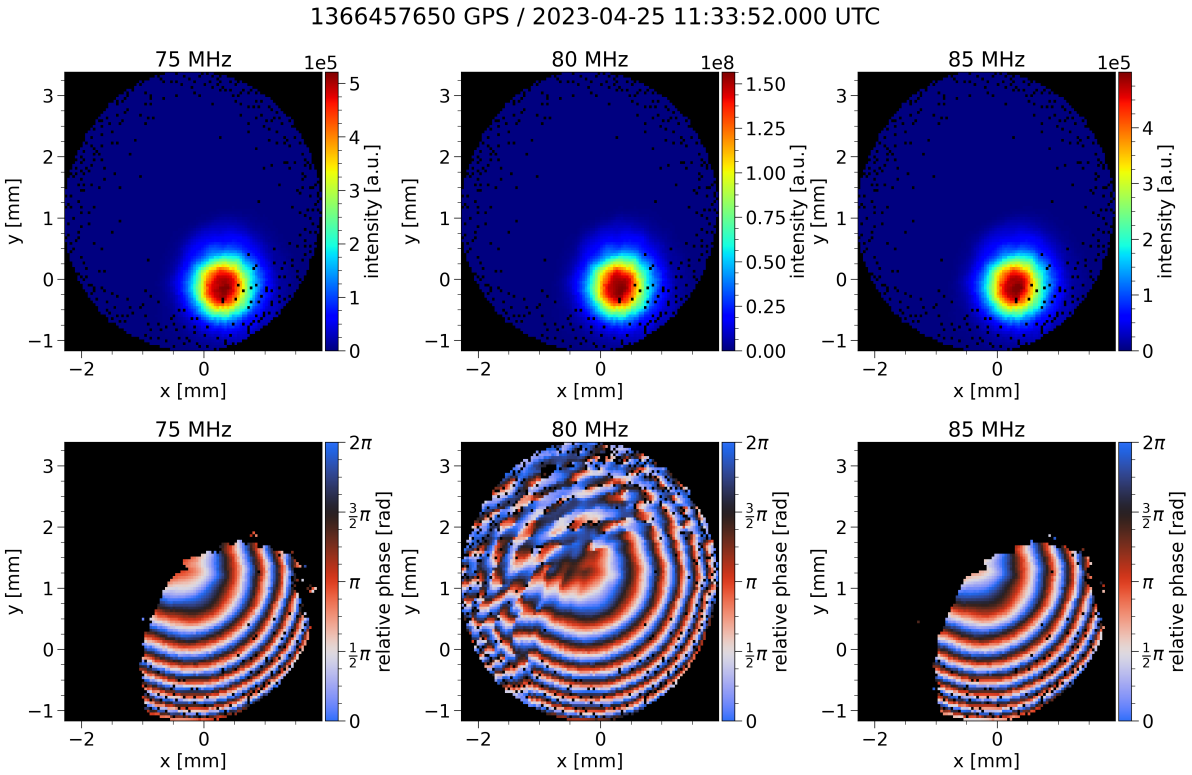


Figure C.7: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.55$ cm. The phase images are centered to measure the phase difference along the scanning radius.

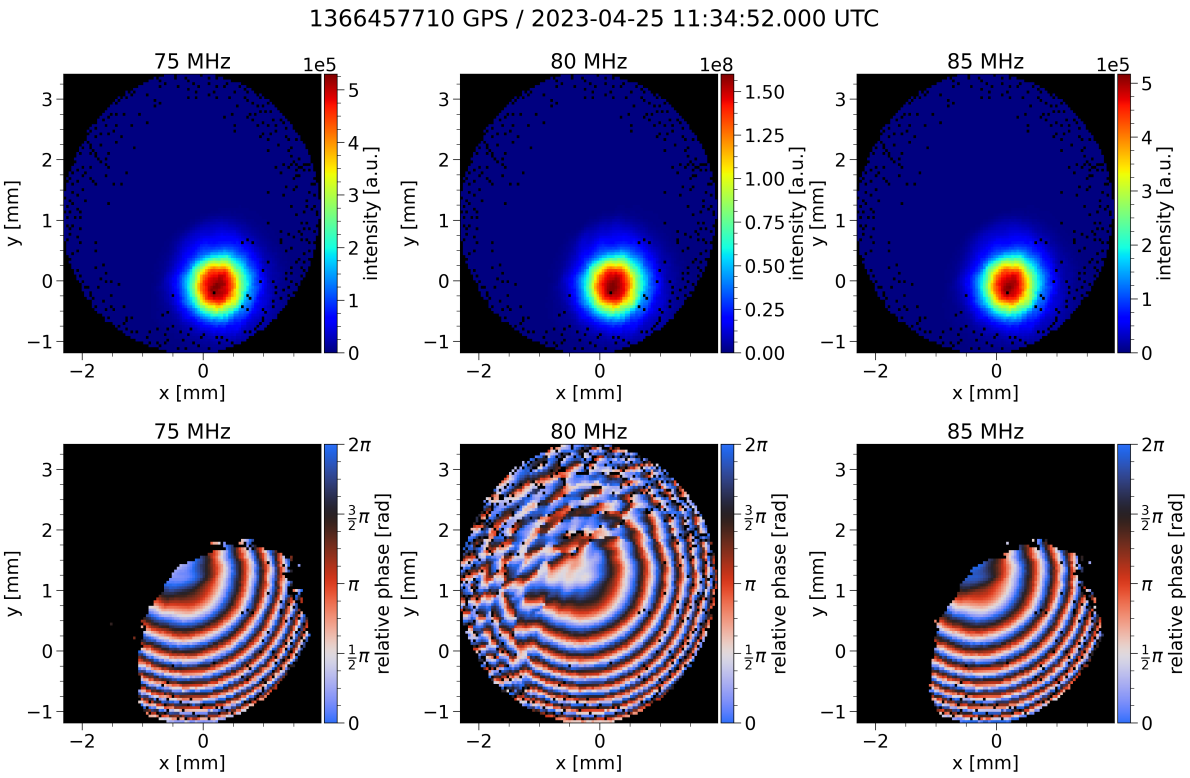


Figure C.8: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.50$ cm. The phase images are centered to measure the phase difference along the scanning radius.

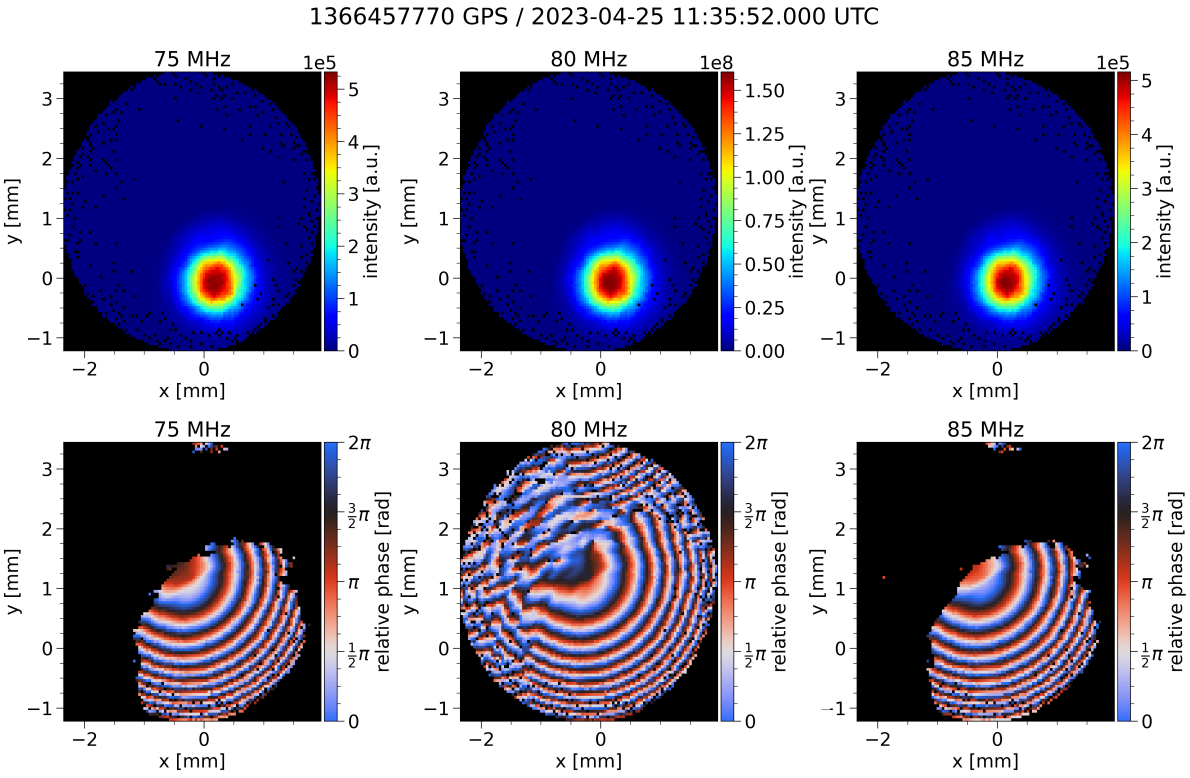


Figure C.9: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.45$ cm. The phase images are centered to measure the phase difference along the scanning radius.

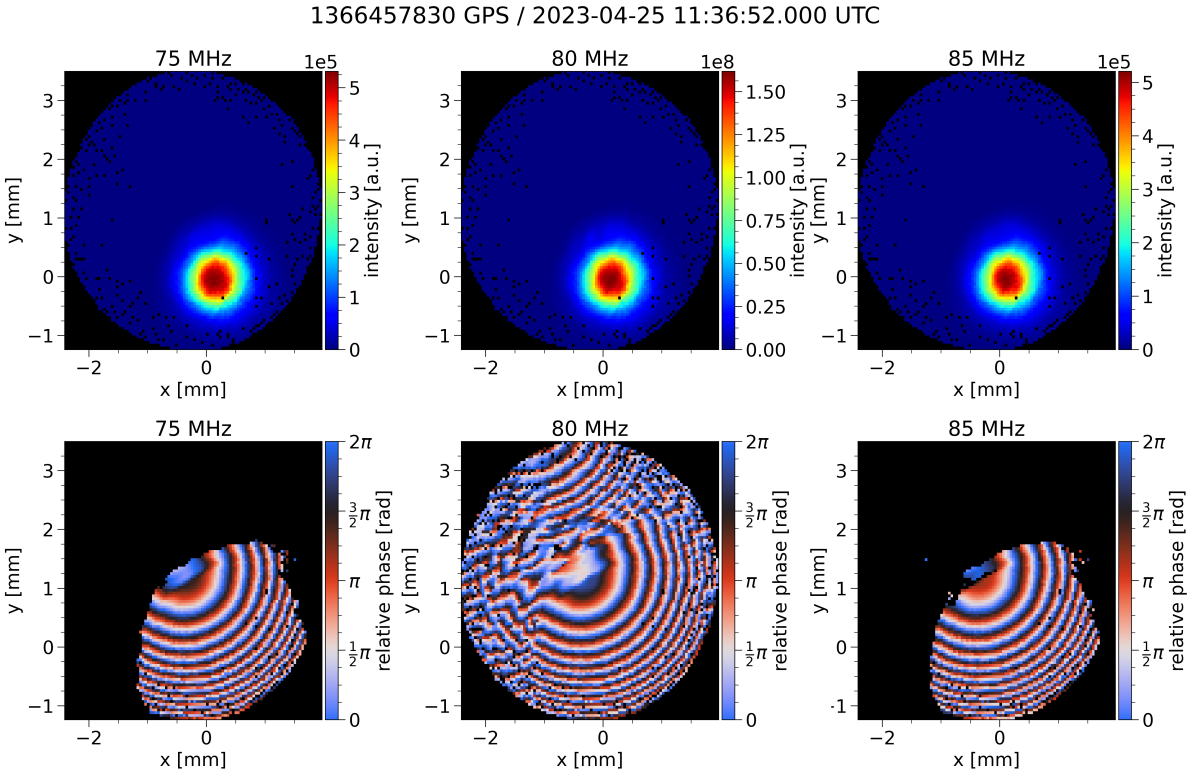


Figure C.10: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.40$ cm. The phase images are centered to measure the phase difference along the scanning radius.

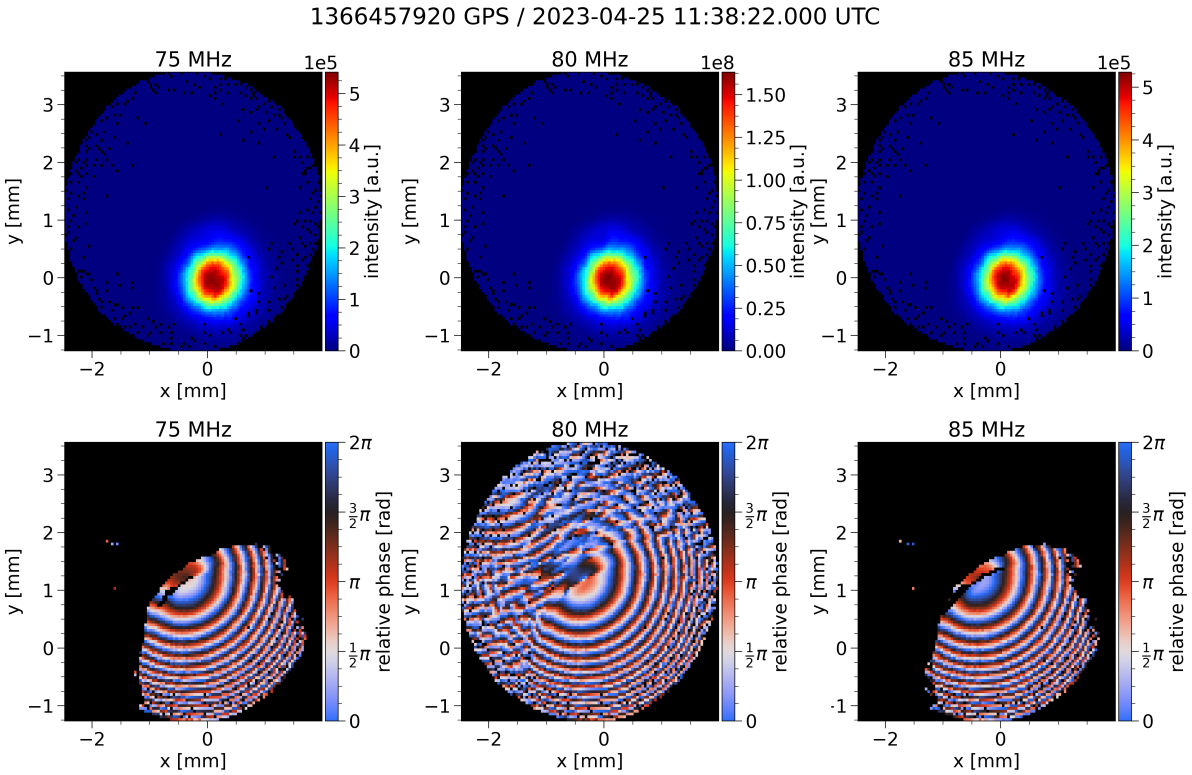


Figure C.11: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.35$ cm. The phase images are centered to measure the phase difference along the scanning radius.

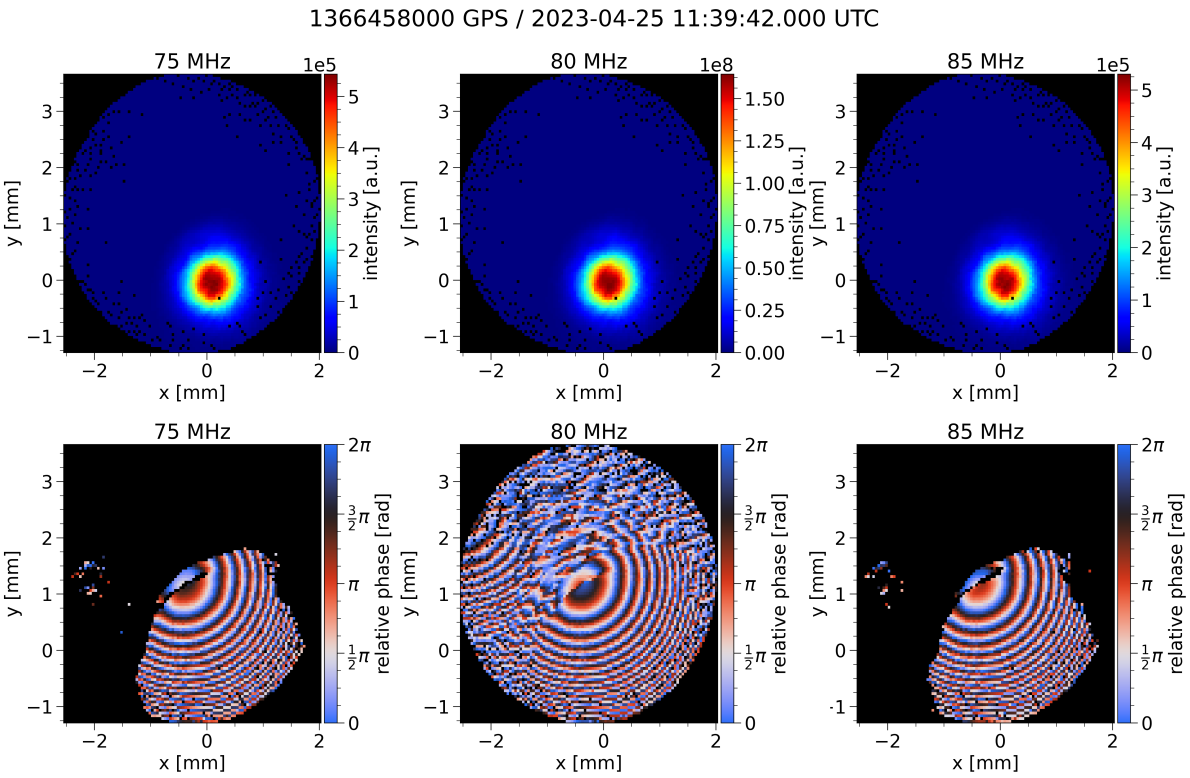


Figure C.12: Phase image to measure the relation between the number of rings and L_2. For this image $L_2 = 0.30$ cm. The phase images are centered to measure the phase difference along the scanning radius.


Nikhef


TU Delft