UPPSALA
UNIVERSITET

21013
Examensarbete 15 hp
Juni 2021

Experimental characterization of focal ratio degradation of optical fibers due to various coupling technologies

Julia Dahlberg
Isabella Rudengren

UPPSALA
UNIVERSITET

Teknisk- naturvetenskaplig fakultet
UTH-enheten

Besöksadress:
Ångströmlaboratoriet
Lägerhyddsvägen 1
Hus 4, Plan 0

Postadress:
Box 536
751 21 Uppsala

Telefon:
018 – 471 30 03

Telefax:
018 – 471 30 00

Hemsida:
http://www.teknat.uu.se/student

Abstract

Experimental characterization of focal ratio degradation of optical fibers due to various coupling technologies

Julia Dahlberg and Isabella Rudengren

The goal of this project was to develop a measuring method and software code to determine and compare the focal ratio degradation of optical fibers for two different coupling technologies. One of the couplings used a fusing technology to splice the fiber, and the other coupling used a refractive index matching technology. Also, an optical fiber without any cleaving or splicing was used as a reference. A collimated beam test was developed as a method for measuring the focal ratio degradation for these different fiber couplings, and a software code was developed to process the results of the experiment. Using the collimated beam test and software code, the focal ratio degradation was calculated and compared between the couplings, and the results clearly stated that the reference fiber had the least focal ratio degradation. The fusing technology used for splicing the fiber had in comparison the least focal ratio degradation of the two different coupling technologies. The results were as expected and therefore the developed measuring method and software were estimated to have been carried out successfully. However, improvements to the measuring method and parts of the software could be done, especially regarding the background light which was a substantial source of error. In conclusion, the goal of the project was reached.

Handledare: Nikolai Piskunov and Alexis Lavail
Ämnesgranskare: Natalia Ferraz
Examinator: Martin Sjödin
ISSN: 1401-5757, UPTEC F21 013

Populärvetenskaplig sammanfattning

Inom astronomisk forskning används ofta optiska fibrer som informationslänk mellan teleskop och mätinstrument, eftersom de bidrar till att mätningar blir mer tillförlitliga då vibrationer och yttre miljöfaktorer får en mindre påverkan på resultaten. Inom detta projekt undersöktes ett par olika kopplingar av optiska fibrer och deras ljusförluster inom benämningen FRD (Focal ratio degradation). En mätmetod utvecklades tillsammans med en mjukvara som användes för nödvändiga beräkningar och analyser som ledde till att FRD kunde bestämmas. Metoden som användes kallas collimated beam test och innebar att parallellt ljus skickades in i ena fiberänden, vilket genererade en ring av ljus från den andra fiberänden. Mätningarna utfördes på tre olika fiber: en fiber var klyvd och sedan sammansmält, en annan fiber var klyvd och sedan ihopkopplad med hjälp av en vätska av matchande refraktionsindex och den tredje fibern var inte klyvd och fungerade som referens. Resultaten givna av mätningarna och mjukvaran var som man hade förväntat sig; FRD var minst för referensfibern och som högst för fibern med en koppling där refraktionsindex-matchande vätska användes. Även om det finns goda möjligheter att utveckla och förbättra mätmetoden, påvisades ett godtagbart resultat och målet med projektet ansågs därför vara uppnått.

Contents

1 Introduction

Revolutionary astronomical discoveries have been made by telescopes around the globe over the past decade, including the finding of new planets and the first imaging of a black hole. These discoveries have contributed to a greater understanding of the universe, and astronomers are now looking to make new astounding discoveries and further enhance their knowledge within astrophysics. To accomplish this, the largest telescope ever built is currently being constructed in Chile, called the Extremely Large Telescope (ELT). The ELT will gather light from astronomical objects and transport it to various instruments where the light will be analyzed [1]. One of these instruments is the spectrograph HIRES that will analyze all the component wavelengths of the light [2]. The light will be transported to HIRES using optical fibers, which will be delivered by Uppsala University. These fibers enable delicate instruments such as HIRES to be mounted on stationary platforms that are in a stable and controlled environment. The instruments are then protected from the turbulent environment from the telescope and its surroundings [3].

The coupling of optical fibers is essential to minimize light losses and achieve an image of desirable quality. Different coupling technologies generate different amounts of light losses, which affects the quality of the image. This project will focus on developing a measuring method and software to characterize the light losses in optical fibers by measuring and comparing the focal ratio degradation between two different optical fibers with different coupling technologies: one which probably will be used in the ELT, where the fiber core is coupled via fusion, and one where the fiber core is coupled using refraction index matching. These will in turn be compared to a fiber functioning as a reference. Hence, the goal of the project is to develop a method to determine the focal ratio degradation for these different optical fibers and analyze the results.

2 Theory

Optical fibers consist of a bundle of thin, cylindrical fibers made of plastic or glass [4]. These materials have the appropriate characteristics, such as flexibility and low attenuation of light, which are advantageous in astronomical applications. One example is transmission of light between telescopes and astronomical instruments where optical fibers enable the possibility to not have direct mechanical coupling between the instruments. This enables flexibility as well as reducing instruments expenses and other complications which can cause light losses [3]. The optical fiber consists of a few layers with different purposes, which are described by Crisp et.al. [5]. The core, the innermost layer, is a transparent and approximate homogeneous layer where the propagation of light mainly occurs. Surrounding the core is a layer called cladding, which has a lower index of refraction than the core. It partly protects the innermost layer, but more importantly, prevents the light from escaping the fiber. For mechanical protection, the fiber also has one or several outermost layers called a buffer or coating which often consist of soft plastic. The core and the cladding are often made of glass, and in typical astronomical application they together have a diameter of approximately 200μm. An illustration of the layers can be seen in figure 1.

Optical fibers are an application of total internal reflection in accordance with Snell's law, which describes the behavior of an electromagnetic wave when it propagates through two different mediums. For total internal reflection to apply within the fibers, the angle of incidence has to be larger than the critical angle [4]. This enables the light rays to propagate along the fiber by a succession of total internal reflections against the coated cladding [6]. The maximum injection angle θ_m for total internal reflection to apply is called the acceptance cone. Directly related to the acceptance cone is the numerical aperture (NA) of the fiber, which is a measurement of the optical fiber's light-gathering ability. By using Snell's law, the numerical aperture of an optical fiber can be written as:

$$N.A. \equiv n_{SRI} \sin\theta_m = n_{core} \cos\theta_c = \sqrt{n_{core}^2 - n_{clad}^2} = \frac{1}{2 \cdot f_{number}} \tag{1}$$

where n_{SRI} is the index of refraction for the medium surrounding the fiber, θc is the critical angle of the fiber, n_{core} is the index of refraction for the fiber core, and n_{clad} is the index of refraction for the coated cladding. The numerical aperture can directly relate to the focal ratio or f-number (which also is a measurement of the light-gathering capability [4]) according to equation (1) [6, 7]. The equation applies for step-index fibers, which are characterized by the discontinuous change in the index of refraction between the core and the cladding in the fibers. There also exist another type of fibers called gradient index fibers, which are characterized by a decreasing index of refraction in the core as the radial distance from the core axis increases [3].

According to Heacox et.al [3] an optical fiber can only propagate a finite number of modes, where modes are described as the eigensolutions to Maxwell's equation on a cylindrical geometry with boundary conditions. A mode can therefore be seen as a wave propagating along the fiber axis with a standing wave as a radial component. Therefore, optical fibers are classified according to the number of modes that can propagate through it [3]. Multi-mode fibers, which are used in this project, are fibers that can propagate multiple modes. Single-mode fibers are another class of fibers, but are not relevant in this project [5].

2.1 Light losses

Since the optical fibers used in applications are not ideal, the propagating light attenuates and spreads due to loss mechanisms, which either can be extrinsic or intrinsic losses. Extrinsic losses involve i.a. geometric effects and inhomogeneities of dimensions larger than the optical wavelength, and among intrinsic losses is mainly Rayleigh scattering [6]. These types of losses also contribute to spreading of the light and are further explained below.

Heacox et. al. [3] categorize extrinsic losses of geometric effects as either macrobending or microbending. Both of these geometric effects cause the requirements for total internal reflection to no longer be fulfilled, and light radiates out of the fiber through refraction. Microbending is defined as faults in the shape of the order smaller than or equal to the core radius, and the source of these faults is mainly due to stress at the interface between the core and the cladding [3]. The stress can for example arise if the fiber gets too cold and the coating becomes smaller. With the condition that the core or the cladding shrinks at a slower rate than the coating, a bend in the fiber might occur and cause a microbend [5]. Macrobending is defined as faults in the shape larger than the order of the core radius of the fiber [3], which involves sharp bends of the optical fiber that causes the angle of the propagating light to change [5]. As previously mentioned, inhomogeneities are also a kind of extrinsic loss. Inhomogeneities can be caused by e.g. an imperfect interface between the core and the cladding or that the fiber material has not been sufficiently mixed before it solidifies [6].

Rayleigh scattering, which is categorized to the intrinsic losses, is due to microscopic inhomogeneities, such as disparities in the index of refraction of the core material or fluctuations in the density of the fiber [6]. The dimensions of these inhomogeneities are of a scale smaller than the wavelength of the light and can be created from, for instance, deficient mixing of the fiber material [5]. There can also be other types of scatterings, such as scattering at the fiber end faces or the interface between the core and the cladding due to surface roughness [6, 8].

Cleaving of an optical fiber also contributes to extrinsic light losses, which is partly because of the gap loss and partly because of Fresnel losses. The gap loss occurs since light exiting the fiber end will spread out according to the acceptance cone and therefore less light will strike the fiber core at the receiving fiber end, which results in light losses [5]. Fresnel losses occur since the light is not only refracted into and out of the fiber ends but also partially reflected. A portion of the light is reflected back towards the light source, which results in an attenuation of the light [9]. These losses can be reduced by different coupling technologies (also known as splicing), by for example filling the gap between the fiber ends with an index matching fluid or by fusing the fiber ends together. The index matching fluid and the fiber core have similar refractive indexes,

which for the light makes the fiber core appear continuous. The same principle is used for the fused fiber, where the fusing makes the fiber core appear continuous to the light. However, the light perceives different amounts of continuity in the fiber core depending on which splicing method is used, which results in different amounts of light spread [5].

2.2 Focal ratio degradation

When performing a collimated beam test on an optical fiber - as done in this project - monochromatic parallel light is sent into an optical fiber with a specific input angle, θ_{in}. A conical light beam with a specific output angle θ_{out}, will emerge from the output end of the fiber due to azimuthal scrambling (which is common for step-index fibers [3]). It will form an annulus (ring pattern) with a specific radius and width (see figure 1).

Due to the light losses described in section 2.1, spreading of the light occurs and results in decreasing of the focal ratio at the output end in comparison with the focal ratio at the input end. This phenomenon is called focal ratio degradation (FRD) and implies that the annulus, projected from the exiting light, has an expanded width [10]. If the fiber would be ideal the annulus would be infinitely thin, thus, the width of the ring is a measurement of the FRD [10]. The FRD can be calculated according to equation (2), where θ_{exp} is the divergence angle or expansion angle of the ring and θ_{out} is the output angle of the cone.

$$\text{FRD} = \frac{\theta_{exp}}{\theta_{out}} \tag{2}$$

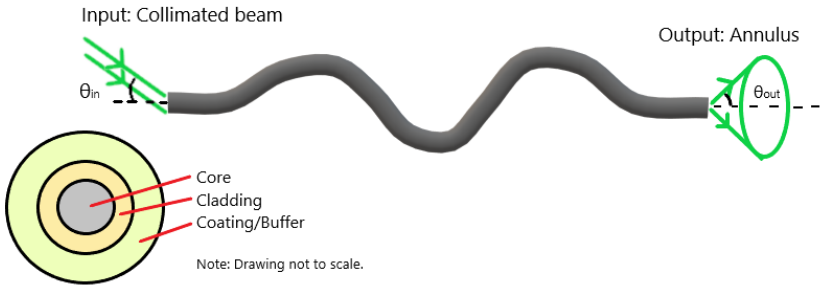


Figure 1: Illustration of collimated beam test and the different layers of the optical fiber.

2.3 Derivation

To calculate the FRD based on collimated beam measurements it is necessary to make two different measurements for each injection angle and fiber type, where the distance from the output end of the fiber to the camera differs. The distance between these two positions (Δd in figure 2) has to be known, as well as the radii (r_1 and r_2) and the widths of the rings (w_1 and w_2). To calculate the output angle of the cone, named θ_{out} in figure 2, which is of interest, one can use the principle of two similar orthogonal triangles, as seen in equation (3). Combining this equation with the expression for the tangent of θ_{out} gives us the final expression for the output angle seen in equation (4).

$$d = \frac{r_1 \Delta d}{r_2 - r_1} \tag{3}$$

$$\theta_{out} = \arctan \frac{r_1}{d} = \{\text{eq.3}\} = \arctan \frac{r_2 - r_1}{\Delta d} \tag{4}$$

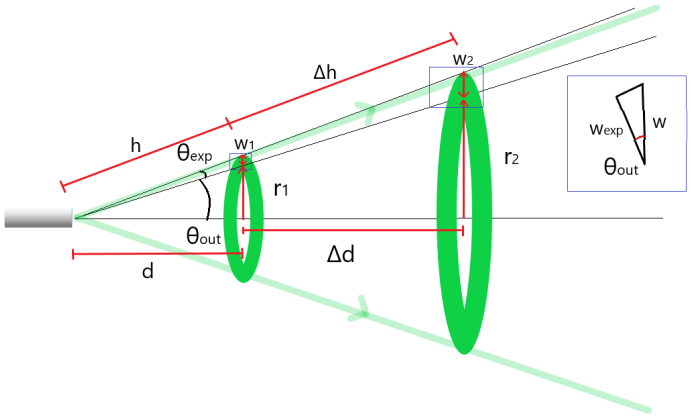


Figure 2: Illustration of output annulus with defined angles and distances.

The divergence angle, also called the expansion angle of the cone wall, can be derived in a similar way. However, this angle is derived from the smaller triangles that can be seen in the blue box in figure 2. In this triangle, w corresponds to w_1 and w_2 which are half of the widths of the rings, and w_{exp} corresponds to the distance of the expansion of the cone wall. These two triangles are also similar triangles, which leads to the expression in equation (5). As before, equation (5) together with the tangent of θ_{out} results in the expression for the angle of interest seen in equation (6).

$$h_1 = \frac{w_{1,exp} \cdot \Delta h}{w_{2,exp} - w_{1,exp}} \tag{5}$$

$$\theta_{exp} = \arctan \frac{w_{1,exp}}{h_1} = \{\text{eq. } 5\} = \arctan \frac{w_{2,exp} - w_{1,exp}}{\Delta h} \tag{6}$$

However, the distances $w_{1,exp}$ and $w_{2,exp}$ which describes the distance the cone wall expands, and Δh in equation (6), is not known. An expression for $w_{1,exp}$ and $w_{2,exp}$ can be formulated if one approximates the smaller triangle, found in the blue box in figure 2, as orthogonal. The resulting expression for the cone wall expansion can be found in equation (7). The expression for Δh can be found by using the cosine expressions for θ_{out} using the larger triangle consisting of the sides $d + \Delta d$, $h + \Delta h$ and r_2, and the smaller triangle consisting of the sides d, h and r_1. This results in equation (8). Combining equation (6) with equation (7) and (8) gives the final expression for the expansion angle θ_{exp} found in equation (9).

$$w_{exp} = w \cos \theta_{out} \tag{7}$$

$$\Delta h = \frac{d + \Delta d}{\cos \theta_{out}} - h = \left\{ h = \frac{d}{\cos \theta_{out}} \right\} = \frac{\Delta d}{\cos \theta_{out}} \tag{8}$$

$$\theta_{exp} = \arctan \frac{\Delta w \cos^2 \theta_{out}}{\Delta d} \tag{9}$$

The expressions for the output angle of the cone, θ_{out}, and the expansion angle of the cone wall, θ_{exp}, found in equation (4) and (9) are used when calculating the final results.

3 Method

3.1 Experimental setup

The experimental setup used consisted of a LED Driver coupled to a Fiber-Coupled LED, which sent out collimated monochromatic light of the wavelength 530 nm (green). The collimated light sent out from the light source was aligned with a rotating stand scaled with 5° intervals. The input end of the fiber was mounted on top of the rotating stand with the fiber end approximately in the center of rotation. The output end of the fiber was mounted on a stand close to a back-illuminated CMOS Image Sensor (ASI178MM) with a resolution of 6.4 Mega Pixels (3096×2080) as seen in figure 3. The camera was connected to a computer and the software *ASIStudio Image* was used to capture the images of the light from the output end of the fiber. A black cloth was used to cover the setup at the output end of the fiber to minimize background light.

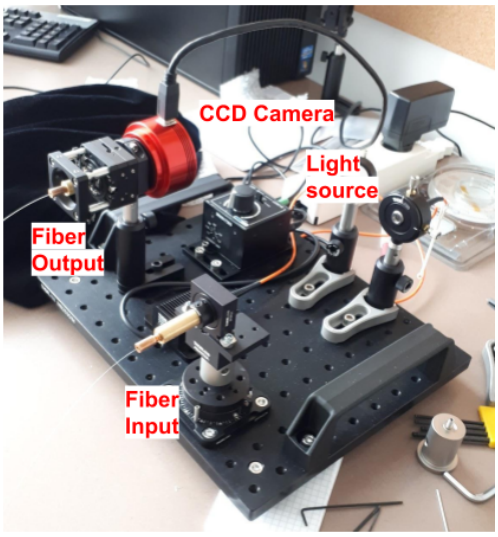


Figure 3: A visualisation of the experimental setup used for measurements.

Both fiber ends were mounted inside a Cu cylinder using two objects made of teflon (see figure 4), which helped to hold the fiber in a fixed position in the cylinder. The fiber ends extend a couple of millimeters out of the cylinder. These were in turn put into a hollow cylinder of brass each, which were mounted on the stands. At the fiber output, the cylinder of brass was mounted on a cage structure which could be moved along the axis of the camera. The movement of the cylinders could be restricted by limiters, and thus different positions of the fiber ends could be set.

3.2 Preparations

During the preparations, a software code for processing the taken images was developed and measurements were performed on an unpolished and partly stressed optical fiber. The test images taken were used during the development of the software code as well as when evaluating the measuring method.

As previously mentioned, the method used during measurements is called collimated beam test and it was performed using the experimental test bench described in section 3.1. Measurements were done for two

Figure 4: Cu cylinder (ferrule) and the teflon fixtures.

different input angles (the values varied), which were adjusted by the rotating stand. The fiber was mounted as close to the CCD camera as possible. A black cloth covered the camera and the output end of the fiber during the measurements to remove some of the background light.

A software code was gradually developed to process the images taken during the test measurements. The code was developed in Python and since the images were saved as FITS-files (Flexible Image Transport System), the package *astropy.io.fits* was used to retrieve the data which consisted of an array with intensity values measured as counts. The goal was to develop a software code that could calculate the radius and the width of the ring received from the output end of the fiber.

The first step was to calculate the center of mass of the image which should be positioned in the center of the ring. This was done by calculating the sum of each data point in the retrieved data array multiplied with the corresponding x- or y-coordinate, depending on which center-coordinate was calculated. The factors of x- and y-coordinates corresponded to the number of the pixel along the x- or y-axis, respectively. To achieve the center of mass, these sums were divided by the sum of the elements in the retrieved data array. This procedure is described by equation (10) where A is the matrix of size (3096×2080) containing the data retrieved from the measurements.

$$\begin{cases} x_{cen} = \frac{\sum_{j=1}^{3096}\sum_{i=1}^{2080} j \cdot A_{ij}}{\sum_{j=1}^{3096}\sum_{i=1}^{2080} A_{ij}} \\ y_{cen} = \frac{\sum_{i=1}^{2080}\sum_{j=1}^{3096} i \cdot A_{ij}}{\sum_{j=1}^{3096}\sum_{i=1}^{2080} A_{ij}} \end{cases} \tag{10}$$

Due to difficulties when determining the center of mass of the test images taken, caused by unwanted background light, the image sometimes had to be cropped before calculating the center of mass to improve the result. This feature was added to the final software code as well as removing an observed hot pixel from the data set.

The next step was to create a 1D cross-section by calculating the radial distance from the center of mass to each pixel and sort the intensity values according to increasing values of the radius. This was done by using the retrieved coordinates of the center of mass to calculate the radii according to the equation of the circle in Cartesian coordinates and then sort it using the in-built method *a.argsort()*. Plotting these values against each other resulted in an approximate Gaussian-shaped curve.

To receive the radius and the width of the ring it was necessary to fit a Gaussian curve to the plotted values (intensity versus radius). Trying to fit a symmetric Gaussian to the curve did not result in a sufficiently satisfactory result and therefore the final software code included the process of fitting an asymmetric Gaussian curve with variable continuum level. Using the method *curve_fit* in the *Scipy*-package along with equation (11) and estimating the parameters A, μ, σ, C_l and C_r - which correspond to the amplitude, the mean value, the standard deviation, the constant on the left side and the constant on the right side respectively - an asymmetric Gaussian curve was fitted to the intensity-versus-radius curve. The coefficients for the curve ware received, including the mean value (equals the radius of the ring) and the standard deviation (used to

calculate the width). The method *curve_fit* uses a built-in Levenberg-Marquardt algorithm[1] to solve the optimization problem.

$$\begin{cases} f(x) = (A - C_l) \exp \frac{-(x-\mu)^2}{2\sigma^2} + C_l & \text{for } x < \mu \\ f(x) = (A - C_r) \exp \frac{-(x-\mu)^2}{2\sigma^2} + C_r & \text{for } x \geq \mu \end{cases} \tag{11}$$

Since the parameters A (amplitude) and μ (mean value) are estimated by finding the maximum intensity value in the array and the corresponding value of the radius, the estimates are very dependent on the spread of the data points in the data set. If there are single data points with a high intensity that is deviating from the rest of the curve, the estimates of these two parameters might not be good enough to find a well-fitted curve. To avoid these situations the in-built function *sigma_clip* was used when needed to mask data points deviating more than five standard deviations. The number of standard deviations used was in rare cases changed to better fit the data set. Worth noting is that the parameters C_l and C_r, functioning as constants, were estimated by calculating the mean value of the thousand leftmost intensities and the thousand rightmost intensities respectively. The parameter σ corresponding to the standard deviation was estimated by hard-coding an approximate value of 70.

The last steps were to calculate the radius and the width of the ring and plot the image with the center of mass and three circles: one where radius equals the mean value, one where the radius equals the sum of the mean value and one standard deviation and one where the radius equals the difference of the mean value, and one standard deviation. Using cylindrical coordinates in 2D the three circles are calculated and plotted. The radius of the ring equals the mean value received from the coefficients of the fitted Gaussian and the width of the ring equals two times the standard deviation of the fitted Gaussian. These values were multiplied with the pixel size of the CCD camera to convert the results from pixels to millimeters.

3.3 Final software code

The final software code, seen in Appendix, can only calculate the results of measurements done with one fiber and input angle at a time. It takes in several inputs; a directory where the FIT-files from the measurements were saved, name of the fiber measured, input angle used, the position of the fiber, date and time for measurements and a directory where figures of plots and a csv-file containing the final results will be saved. The code consists of six functions (*averaging, center_of_mass, cross_section, asym_gauss, fit_gauss, calc_result*) which contain the steps described in section 3.2. Important to note is that the function *averaging* makes an averaged image/data set of the measurements given to reduce the impact of fluctuations.

The software returns the results in the output log or as a plot as well as saved files or figures. The values of the radius and width of the ring in millimeters together with information regarding the measurement processed (which was given as input) can be found in the output log. However, this information is also saved in a csv-file named *ring_data.csv*. If one processes several measurements, all of the results will be saved in the same csv-file. The final plots - which include the final annulus with the center of mass and the three circles, and the graph with the data set and the fitted Gaussian - are saved as PNG-files in the specified directory.

[1] Levenberg-Marquardt algorithm is a useful method for solving least square problems involving nonlinear functions [11].

3.4 Execution

During the execution, measurements were performed on three different multi-mode fibers of type LG105LCA and of the same length ($\sim$ 1 m), with a core of 105 μm, a cladding of 125 μm and a coating of 250 μm. The fibers must be of the same length since FRD is dependent on the fiber length. The three fibers are named "Splice 1", "Splice 2" and "No process 3" and all fibers were cleaved on both ends with an accuracy of $\sim 0.5°$. "Splice 1" and "Splice 2" were cleaved, spliced and recoated by the company Nyfors Teknologi AB. "Splice 1" were spliced using a matching refractive index fluid and "Splice 2" used fusing as splicing. "No process 3" was an unprocessed fiber used as a reference.

A similar experimental setup to the one used during the preparations (seen in section 3.1) was used for the final measurements. The collimated beam test was performed as before with three different input angles: 6°, 8.5° and 11°. However, this time each input angle was measured at two different fixed positions with a distance of 5 mm in between. The two positions were fixed using a caliper and the limiters on the cage structure. The fiber could then be pushed between the two positions. The specific distance between the two fixed positions was chosen considering the out-coming annulus was mostly visible in the image for a larger interval of input angles. The specific interval of angles was chosen considering the numerical aperture of the fiber, which was approximately found by studying the image generated from the output end of the fiber (see figure 5), and making sure the image received was a distinguishable annulus at all input angles and at both fixed positions. The position of the output end of the fiber was adjusted to make sure the ring was approximately in the middle of the image. 100 images were captured for each angle and position (to be able to remove fluctuations), which results in 18 different measurements.

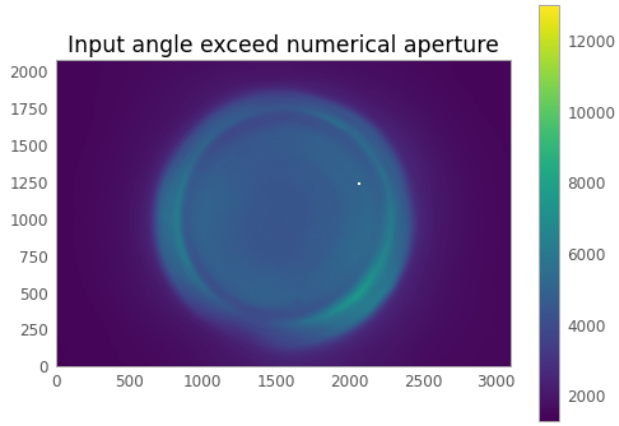


Figure 5: The image received when the input angle exceeds the numerical aperture.

The fit-files gained from the 18 measurements were then processed one at a time, using the software code described in section 3.3, to receive the resulting radii and widths. The calculations were performed manually according to the formulas in section 2.3.

4 Results

The resulting radii and widths of the ring in millimeters for each fiber, injection angle and fixed position can be found in table 1. The position marked 5 mm (in brackets) corresponds to that the fibers were positioned 5 mm further away from the camera compared to the position 0 mm. The results in table 1 are the resulting values when processing the measurements in the software code. As an example, the resulting plots of the ring with the three circles and center of mass together with the corresponding graph showing the fitted Gaussian on the intensity versus radius curve can be found in figure 6. The measurements shown in the plots are solely for measurements done with injection angle 8.5° at the position 0 mm for the three different fibers.

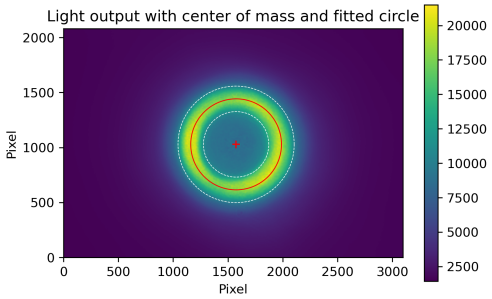
The resulting output angle, expansion angle and FRD for each fiber and injection angle, were calculated by using equation (2), (4) and (9). These values can be found in table 2. The results come from calculations made based on the results in table 1. Overall, the fiber named "No process 3" has the least FRD for each injection angle out of the three fibers, followed by "Splice 2" which has less FRD than "Splice 1", but more than "No process 3". At the same time, all output angles are smaller than the corresponding injection angle.

Table 1: Resulting radii and widths received from processing measurements in software code.

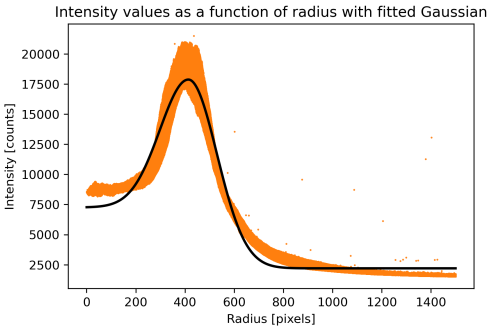
Fiber name	Injection angle	Radius [mm] (0 mm)	Width [mm] (0 mm)	Radius [mm] (5 mm)	Width [mm] (5 mm)
Splice 1	11°	1.3024	0.4065	2.1359	0.6705
Splice 1	8.5°	0.9932	0.5567	1.6326	0.8301
Splice 1	6°	0.6489	0.4258	1.0899	0.7059
Splice 2	11°	1.3520	0.4137	2.2664	0.5693
Splice 2	8.5°	1.0272	0.5395	1.7436	0.7505
Splice 2	6°	0.7109	0.3818	1.1817	0.5861
No process 3	11°	1.3883	0.3038	2.3427	0.4440
No process 3	8.5°	1.0682	0.3633	1.7382	0.5269
No process 3	6°	0.7223	0.5065	1.2069	0.6094

Table 2: Resulting angles and FRD for all fibers and injection angles measured.

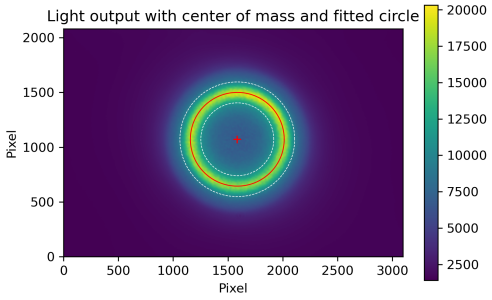
Fiber name	Injection angle	Output angle	Expansion cone wall	FRD
Splice 1	11°	9.46°	1.47°	0.155
Splice 1	8.5°	7.29°	1.54°	0.211
Splice 1	6°	5.04°	1.59°	0.315
Splice 2	11°	10.36°	0.86°	0.083
Splice 2	8.5°	8.15°	1.18°	0.145
Splice 2	6°	5.38°	1.16°	0.216
No process 3	11°	10.81°	0.77°	0.071
No process 3	8.5°	7.63°	0.92°	0.121
No process 3	6°	5.54°	0.58°	0.105



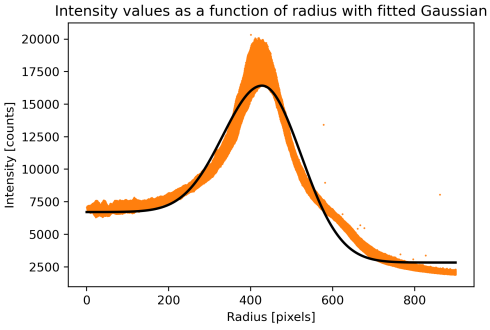
(a) Splice 1: 8.5° and 0 mm



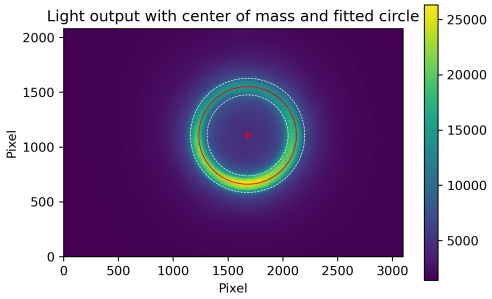
(b) Splice 1: 8.5° and 0 mm



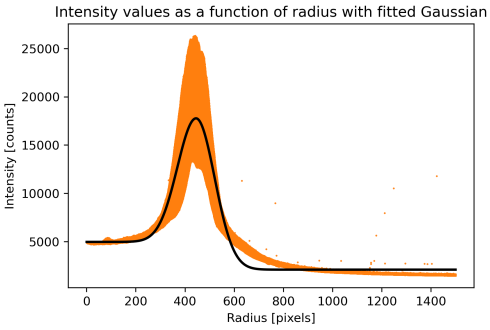
(c) Splice 2: 8.5° and 0 mm



(d) Splice 2: 8.5° and 0 mm



(e) No process 3: 8.5° and 0 mm



(f) No process 3: 8.5° and 0 mm

Figure 6: Resulting images and plots for the different fibers with injection angle 8.5° and position 0 mm.

5 Discussion

The results in table 2 give information about the FRD for the different fibers. As mentioned before, the fiber named "No process 3" has the least FRD, which is as expected. The fiber was not cleaved and spliced and therefore one of the contributions to an increased spread of light is avoided. Only the characteristics of the fiber itself, such as stress and surface roughness, contributes to the FRD. Regarding "Splice 2" and "Splice 1", "Splice 2" have less FRD than "Splice 1" considering all injection angles. This result is also as expected; the fiber "Splice 2" is spliced through fusing which has a smoother transition of refractive index compared to the fiber "Splice 1" which uses a matching refractive index fluid. Therefore the spread of the light should be smaller for "Splice 2", and hence less FRD. One other thing worth noting regarding FRD is the increase of FRD for decreasing injection angles. However, there is one exception: the unprocessed fiber for the injection angles 8.6° and 6°. This can probably be explained by the asymmetric ring and the difficulties with the processing this lead to. This resulted in an uncertain result.

However, there may have been several factors that could have affected the resulting FRD values negatively. First and foremost, there was a lot of background light during the measurements which can be seen in figure 6. During the development of the software code, this factor contributed to a lot of difficulties which demanded the use of manual adjustments to find a feasible value of for example the center of mass and the radii. Several different strategies were tested to still get a good result with the background light such as using thresholds and cropping of the image. The background is also the reason for using an asymmetric Gaussian to fit the plotted curve of the intensities as a function of the radius. Without the background, light it would be more symmetric.

Another factor that may have affected the result is the inaccurate scaling of the rotation stand (5° intervals) and the fact that it was not properly aligned. Due to the small interval of injection angles that was possible to use, we had to use an injection angle between two scaling lines, which probably meant that the injection angle could be at least a couple of tenths of degrees wrong. Also, the fiber was not properly aligned with the light source when the scale on the rotating stand was set to 0°. We had to make an approximation of the 0° position by using the image produced on the camera, which also could have contributed to the uncertainty of the injection angle. Smaller deviations can also come from the collimated light beam not being properly collimated.

Some of the FRD measured could have been caused when handling and mounting the fibers. It was difficult to not stress the fiber ends (which were bare and therefore very fragile) when putting these into small cavities which the mounting procedure demanded (see figure 4).

Future work could consist of increasing the accuracy of the measuring method, as well as make the mounting procedure less delicate. For example, one could use a more accurate rotating stand with a smaller interval of angles that is aligned with the light source. A lot of the uncertainties regarding the injection angle would then be eliminated. It would also be desirable to eliminate the background light as much as possible. One solution could be to do the measurements in a darkened room which would eliminate the background light more effectively than the black cloth. This would probably make the software code more effective and accurate.

During the procedure of processing the measurements, one observation was done. The values of the radius and width of the ring were very sensitive to small changes in the position of the center of mass. Since the position of the center of mass was adjusted when needed by the naked eye and sometimes with respect to the resulting circles' positions, this could lead to larger deviations in both radii and widths. These deviations could have been of less importance in the results presented in this report if more measurements for each angle and fiber had been done. Another way could have been to split the measurements done into smaller data sets and process these instead.

6 Conclusion

A measuring method, the collimated beam method, was successfully developed and a software code was continuously developed alongside the development of the method. The results received from the 18 measurements done on the three different optical fibers were as expected; the fiber functioning as references had least FRD and the fused fiber had less FRD than the fiber using a matching refractive index fluid. Improvement can be done regarding the sources of error, especially referring to the background light. With the background light removed, the results would be more accurate and fewer adjustments to the processing software code would have to be done.

References

[1] ESO. In a nutshell. 2020. URL: https://elt.eso.org/about/ (visited on 05/10/2021).

[2] ESO. HIRES. 2020. URL: https://elt.eso.org/instrument/HIRES/ (visited on 05/10/2021).

[3] William D. Heacox and Pierre Connes. "Optical fibers in astronomical instruments". In: The Astronomy and Astrophysics Review 3 (1992), pp. 169–199.

[4] Hugh D. Young and Roger A. Freedman. University Physics with Modern Physics. 14th ed. Pearson Education, 2016. Chap. Optics, pp. 1106–1112, 1161–1162.

[5] John Crisp and Barry J. Elliott. Introduction to Fiber Optics. 3rd ed. Newnes, 2005.

[6] S.J. Frank L. Pedrotti, Leno M. Pedrotti, and Leno S. Pedrotti. Introduction to Optics. 3rd ed. 2006. Chap. Fiber Optics, pp. 247–253.

[7] R. Haynes et al., eds. Optical Fabrication, Metrology and Material Advancements for Telescopes. Vol. 5494: New age fibers: the children of the photonic revolution. 2004, pp. 586–597. DOI: 10.1117/12.550991.

[8] D. M. Haynes et al. "Relative contributions of scattering, diffraction and modal diffusion to focal ratio degradation in optical fibres". In: Monthly Notices of the Royal Astronomical Society 414.1 (2011), pp. 253–263.

[9] Samira Farsinezhad and Faramarz E. Seraji. "Analysis of Fresnel Loss at Splice Joint Between Single-Mode Fiber and Photonic Crystal Fiber". In: International Journal of Optics and Applications (Feb. 2012), pp. 17–21.

[10] D. Finstad et al. "Collimated focal ratio degradation testing for highly multiplexed fiber systems - an improvement to a standard test". In: Applied Optics 55.25 (2016), pp. 6829–6831.

[11] Eric. W. Weisstein. Levenberg-Marquardt Method. From MathWorld–A Wolfram Web Resource. URL: https://mathworld.wolfram.com/Levenberg-MarquardtMethod.html (visited on 05/12/2021).

Appendix

The final software code used when processing the measurements made.

```
"""
PROCESS FRD MEASUREMENTS OF FIBERS USING COLLIMATED BEAM TEST

This software is used to process the resulting files from FRD measurements
using the collimated beam test of optical fibers.
Note: it can only calculate the result of measurements done with one specific
fiber and input angle at a time.

It takes a folder containing multiple files and calculates an averaged image
which will be further analyzed.

The results contain the radius and the width of the annulus and plots of
the results.

Inputs:
    Directory to the resulting files from measurements
    Name of fiber
    Value of input angle
    Fixed position of fiber when measuring
    Date and time when measurement was performed
    Directory where figures of plots and the results will be saved

Return:
    A csv-file containing the name of the fiber, date and time of measurement,
    input angle, position, radius of the ring and width of the ring.
    Figure (png-file) of plot of annulus with center of mass and three
        circles. The circles have radii mu, mu+sigma and mu-sigma.
    Figure (png-file) of graph where intensities are plotted against radii as
        well as the fitted asymmetric Gaussian curve.
All of these can also be found in the output log and as plots in the Python
software used!

Extra:
    Activate sigma_clip (removes data points outside a specified number of
        standard deviations) on row 251 by removing #. Used when having
        difficulties determining the mean value of the Gaussian (fitting
        curve).
    Change values in array crop in row 256 if having difficulties getting
        a good center of mass.

By: Julia Dahlberg and Isabella Rudengren
Supervisors: Nikolai Piskunov and Alexis Lavail
"""

from astropy.io import fits
from astropy.stats import sigma_clip
import matplotlib.pyplot as plt
import numpy as np
import numpy.ma as ma
import glob
from scipy.optimize import curve_fit
import csv
plt.rcParams['figure.dpi'] = 300    # Improves resolution

def averaging(dirname):
    """Calculates an averaged image of several given images and plots the final
    averaged image.
    Input: directory to folder with files (with '/')
    Output: one averaged image"""
    filenames = glob.glob(dirname+"*fit")
```

```
    AvImg = np.zeros((2080, 3096))

    for filename in filenames:
        with fits.open(filename):
            image_data = fits.getdata(filename, ext=0)

        image_data = image_data.astype('float64')

        AvImg = AvImg + image_data

    AvImg = AvImg/np.shape(filenames)

    plt.imshow(AvImg, cmap='viridis', origin='lower')
    plt.colorbar()
    plt.title('The original image')
    plt.xlabel('Pixel')
    plt.ylabel('Pixel')
    plt.show()
    return AvImg

def center_of_mass(imgdat, crop=[0, 0, 0, 0]):
    """Calculates center of mass of input image, imgdat, and plots it
    together with image.
    Input:
        imgdat - input image
        crop - consist of [bot, top, lef, rig] which will remove number of
           pixels in the given direction. Can be used when the image has to
           be cropped to improve the position of the center of mass.
           Default: [0,0,0,0] not cropping
    Return:
        x_cen - x-coordinate of center of mass
        y_cen - y-coordinate of center of mass"""

    # Crop image if needed
    bot, top, lef, rig = crop
    imgdat_cen = imgdat[bot:(2080-top), lef:(3096-rig)]

    # Get the center of the circle by computing the center of mass
    size = np.shape(imgdat_cen)
    # Create pixel coordinates
    x = np.arange(1, size[1]+1, dtype='int64')
    y = np.arange(1, size[0]+1, dtype='int64')

    x_cen = np.sum(x*imgdat_cen)/np.sum(imgdat_cen)+lef
    y_cen = np.sum(y.reshape(size[0],1)*imgdat_cen)/np.sum(imgdat_cen)+bot

    # Plot (cropped) image with center of mass
    plt.rcParams["axes.grid"] = False
    plt.imshow(imgdat_cen, cmap='viridis', origin='lower')
    plt.colorbar()
    plt.plot(x_cen-lef, y_cen-bot, '*', color='red')
    plt.title('Image with center of mass')
    plt.xlabel('Pixel')
    plt.ylabel('Pixel')
    plt.show()

    return x_cen, y_cen

def cross_section(imgdat, x_cen, y_cen, ):
    """Create the 1D cross-section; calculates radii according to circle's
    equation and sort the intensity values by increasing radius.
    Input:
        imgdat - image
        x_cen - x-coordinate center of mass
        y_cen - y-coordinate center of mass"""
```

```
    size = np.shape(imgdat)
    r_cross = np.empty(size)

    x = np.arange(1, size[1]+1, dtype='int64')
    y = np.arange(1, size[0]+1, dtype='int64')

    xs = (x-x_cen)**2
    ys = (y-y_cen)**2
    r_cross = np.sqrt(np.tile(xs,(size[0],1)) + np.tile(ys.reshape(size[0],1), (1, size[1]))
    )

    # Sort r_cross and image_data in ascending order of r_cross
    couple = ma.column_stack((r_cross.reshape(size[0]*size[1]), imgdat.reshape(size[0]*size
    [1])))
    rf_cross_sort = couple[couple[:,0].argsort()]

    return rf_cross_sort

def asym_gauss(x, *p):
    """Asymmetric Gaussian function with variable continuum level.
    Input:
        x - x-values (radii)
        *p - parameters [Amplitude, mean value, standard deviation,
                         constant left, constant right]
    Output:
        f - y-values, fitted values of assymmetric Gaussian"""
    A, mu, sigma, Kl, Kr = p
    f = np.zeros(len(x)) # initializing the array with the result
    cut = np.argmax(x > mu) # find x corresponding to the mean value
    f[0:cut] = (A-Kl)*np.exp(-(x[0:cut]-mu)**2/(2.*sigma**2)) + Kl
    f[cut:] = (A-Kr) *np.exp(-(x[cut:]-mu)**2/(2.*sigma**2)) + Kr
    return f

def fit_gauss(rf_cross_sort, radius=1500):
    """Fits an asymmetric Gaussian function to the intensity versus radius
    curve and plots the fitted curve together with the data. Saves plot as png
    in given directory.
    Input:
        rf_cross_sort - data set containing intensity and radius values
        radius - remove data points larger than the given radius when fitting
                 Gaussian curve. Default=1500
    Output:
        coeff - coefficients for the fitted Gaussian: [amplitude, mean value,
                standard deviation, constant left, constant right]
        png of graph with data and fitted Gaussian"""
    # Remove datapoints larger than the specified radius
    rf_cross_sort = rf_cross_sort[rf_cross_sort[:,0] <= radius]
    # Define parameters for assymmetric Gaussian [amplitude, mean,
    # standard deviation, constant on the left, constant on the right]
    p0 = [np.max(rf_cross_sort[:,1]), rf_cross_sort[np.argmax(rf_cross_sort[:,1]),0], 70,
      np.mean(rf_cross_sort[0:1000,1]), np.mean(rf_cross_sort[-1000:-1,1])]
    # Fit curve
    coeff, var_matrix = curve_fit(asym_gauss, rf_cross_sort[:,0] ,rf_cross_sort[:,1], p0=p0)

    ffit= asym_gauss(rf_cross_sort[:,0], *coeff) # Fitted values

    # Plot graph with assymmetric Gaussian
    plt.plot(rf_cross_sort[:,0], rf_cross_sort[:,1], '.', markersize=1, color="C1")
    plt.plot(rf_cross_sort[:,0], ffit, 'k', linewidth=2)
    plt.xlabel('Radius [pixels]')
    plt.ylabel('Intensity [counts]')
    plt.title('Intensity values as a function of radius with fitted Gaussian')
    plt.savefig(pathsave + 'graph_'+ fiber_name + '_' + input_angle + '_' + dist
                + '_' + date_time + '.png', bbox_inches='tight')
    plt.show()
```

```
    return coeff

def calc_result(imgdat, coeff, x_cen, y_cen):
    """Calculates the radius and the width of the ring and plots the final
    image with center of mass and three circles with radii mu, mu+sigma and
    mu-sigma. Save plot in given directory.
    Input:
        imgdat - data set containing the image
        coeff - coefficients of the fitted Gaussian
        x_cen - x-coordinate for center of mass
        y_cen - y-coordinate for center of mass
    Return:
        mu -  radius of ring (mu)
        width - width of ring (2*sigma)
        Plot of ring with center of mass and circles"""
    mu = coeff[1]
    sigma = coeff[2]

    # Plots with circle and center of mass
    angles = np.linspace(0, 2*np.pi, 200)
    # Best-fit circle
    x_data = mu*np.cos(angles)
    y_data = mu*np.sin(angles)

    # +1 sigma circle
    x1 = (mu+1*sigma)*np.cos(angles)
    y1 = (mu+1*sigma)*np.sin(angles)

    # -1 sigma circle
    x2 = (mu-1*sigma)*np.cos(angles)
    y2 = (mu-1*sigma)*np.sin(angles)

    width = 2*sigma   # Width of circle

    plt.imshow(image_data, cmap='viridis', origin='lower')
    plt.plot((x_data+x_cen), (y_data+y_cen), color='red', linewidth=0.7)
    plt.plot((x1+x_cen), (y1+y_cen), color='white', linestyle="dashed", linewidth=0.5)
    plt.plot((x2+x_cen), (y2+y_cen), color='white', linestyle="dashed", linewidth=0.5)
    plt.title('Light output with center of mass and fitted circle')
    plt.xlabel('Pixel')
    plt.ylabel('Pixel')
    plt.colorbar()
    plt.plot(x_cen, y_cen, '+', color='red')
    plt.savefig(pathsave + 'fig_'+ fiber_name + '_' + input_angle + '_' + dist
                + '_' + date_time + '.png', bbox_inches='tight')
    plt.show()

    mu, width = mu*pix_size, width*pix_size
    return mu, width

# Ask for inputs
dir_name = input('State directory, end with / (example: C:/path/to/measurements/): ')
fiber_name = input('State name of fiber: ')
input_angle = input('State value of input angle [degrees]: ')
dist = input('State fixed position of the fiber [mm]: ')
date_time = input('State date and time for measurement on form YYYY-MM-DD HH_MM_SS : ')
pathsave = input('State path to save plots, end with / (example: C:/path/to/save/): ')

pix_size = 2.4e-3   # Size of pixels camera [mm]

image_data = averaging(dir_name)    # Averaging images

# Mask hotpixel
hot_p = np.max(image_data)
```

```
image_data = ma.masked_equal(image_data, hot_p)

# Can be used to improve the result of the radius:
#image_data = sigma_clip(image_data, sigma=5) # Remove data points outside 5 stddev

x_cen, y_cen = center_of_mass(image_data, [0, 0, 0, 0]) # Center of mass
rf_cross_sort = cross_section(image_data, x_cen, y_cen) # 1D cross-section
coeff = fit_gauss(rf_cross_sort)    # Coefficients from fitted Gaussian
mu, width = calc_result(image_data, coeff, x_cen, y_cen)    # Radius and width

# Print and save results as csv-file in same folder as software code
print(f'{fiber_name} [{date_time}]')
print('--------------------------------')
print(f'Input angle:    {input_angle} degrees')
print(f'Position:       {dist} mm')
print(f'Circle radius:  {"%.4f" % mu} mm')
print(f'Circle width:   {"%.4f" % width} mm')

with open(pathsave + 'ring_data.csv', mode='a') as csvfile:
    fields = ['Fiber name', 'Date&Time', 'Input angle [degrees]',
              'Position [mm]','Radius [mm]','Width [mm]']
    writer = csv.DictWriter(csvfile, fieldnames=fields)
    writer.writeheader()
    writer.writerow({'Fiber name': fiber_name, 'Date&Time': date_time,
                     'Input angle [degrees]': input_angle,
                     'Position [mm]': dist, 'Radius [mm]': {"%.4f" % mu},
                     'Width [mm]': {"%.4f" % width}})
```