Design and Optimization of fiber couplers using Ray Tracing

by

Chatzikostas Antonis

University Of Crete

Supervisor:
Dimitris Papazoglou

Committee Members:
Mr. Petros Rakitzis
Mr. Konstantinos Makris

September 2023

Abstract

Fiber couplers serve as crucial components in modern optical systems, facilitating the efficient coupling of light into and out of an optical fiber. This thesis delves into the world of fiber collimators, exploring their fundamental principles, different designs, aberrations, and the critical role they play in optical fiber applications.

The thesis explains the basics principles guiding fiber collimators, shedding light on their design, starting with a thorough investigation of the fundamentals. The adventure continues into the world of geometrical optics, where optical phenomena are clarified and help to understand how light behaves when passing through an optical system. This thesis examination of the aberrations present in fiber collimation systems is a key component. This research analyzes the aberrations that might impair optical performance and discusses methods to lessen and fix them, improving the accuracy and dependability of optical coupling. The study also explores the world of optical fibers, clarifying their properties, divisions, and useful mechanics.

The use of sophisticated software tools for thorough analysis and optimization of fiber coupling systems is also explored in this thesis. Modern software "Optica" helps with the simulation, modeling, and testing of collimators, allowing engineers and researchers to create designs with high levels of efficiency and accuracy.

Additionally, this thesis introduces two developed fiber coupling systems that were made with authentic Thorlabs optical fibers. These practical applications highlight the seamless fusion of theory and practice and show the direct relevance of the knowledge acquired in this thesis to real-world settings.

Acknowledgements

I would like to express my sincere gratitude to all those helped me throughout the journey of completing this under graduate thesis.

First and foremost, i am deeply thank full to my thesis advisor, Mr.Dimitris Papazoglou for his invaluable guidance, support and time he sacrificed training me and transforming my perspective from that of a student to that of a engineer, eager to deal with problems and focused on the tasks at hand.

My thanks also go to my family for their love, patience and support they have given to me during this thesis.

I would also like to thank my friends who also provided me with positive energy in this crucial period.

Contents

Chapter 1

Introduction

1.1 General Principles

In everyday life, we often use the term "light" to refer to the radiation that is being detected by our own eyes, meaning the visible spectrum. In Physics though, light is a much more complicated term to understand. The electromagnetic spectrum ranges from cosmic rays to radio waves. The nature of each type of electromagnetic radiation can vary widely. Each type of light radiation can travel at the same speed in the vacuum, with a velocity value of $c = 2.998 * 10^{10}$ cm /sec.

We can define the term light wave propagation as the transfer of the energy produced by an electromagnetic wave from one point to another. Now, let's consider a point source in the vacuum of space that emits light. That point source creates a wave front in space in all directions which is spherical in shape. The following wave front propagation through space can be seen in Fig.1.1.

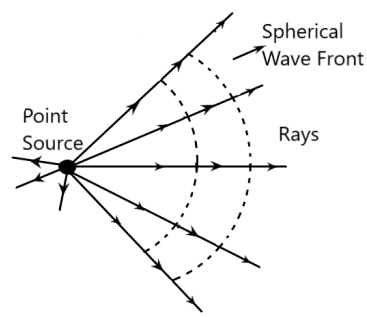


Figure 1.1: Point source that emits light and creates spherical waves.

In other media, the velocity of the waves is different compared to the vacuum one. We define the index of refraction, as the ratio of the velocity in vacuum to the velocity in the medium

$$\text{Index of refraction N} = \frac{\text{velocity in vacuum}}{\text{velocity in medium}} \tag{1.1}$$

Snell's Law

The wavefront of the light source changes when the light enters another medium with a different value of the index of refraction. Now in order to visualize this deformation, let's consider a plane wave that hits another medium at an angle, showed in Fig.1.2. The wave

front goes from a medium with an index of refraction n1 to a medium with a value n2, with n2 greater than n1.

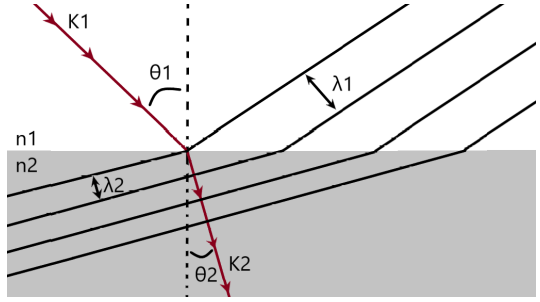


Figure 1.2: A plane wave front passing through one medium to another one.

According to Snell's law, the ratio of the sines of angle of incidence(θ_1) and the angle of the refraction(θ_2) is equal to the ratio of the second refraction index n2 and the refractive index n1. This can be described by the following equation :

$$n_1 sin\theta_1 = n_2 sin\theta_2 \tag{1.2}$$

Taking into consideration the Snell's law of refraction, we can analyse the light passing through surfaces, such as planoconvex and planoconcave lenses, prisms and wedges.

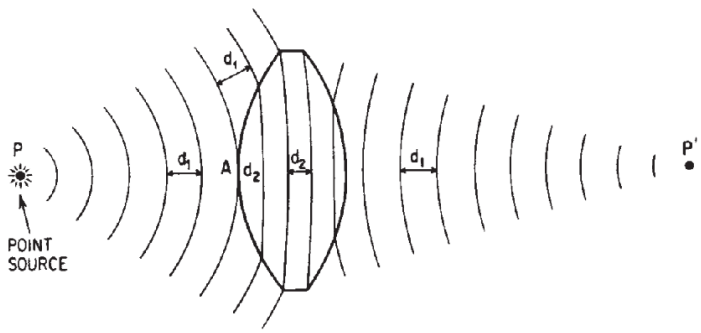


Figure 1.3: Wave fronts passing through a lens[12, Page 15]

Now, let's consider a light point source that emits light and a converging plano lens element with a different medium, as seen in Fig.1.3. The rays propagate through air until they meet this surface. The lens has a higher index of refraction than air. Using the relation in Eq.1.1, the velocity of the wavefront travelling through the lens is lower. As a result,the distance between each wavefront is smaller than the one they have outside of the lens. The same process is happening but in reverse as the rays exit the lens. For this reason, the light from the P point of the system is diverging before entering the component, while exiting it, it converges. The same principles apply to all different components mentioned, depending on the use of the optical system.

Total Internal Reflection

Using the Eq.1.2 we can assume that $\theta_2 > \theta_1$ for $n_1 > n_2$. As the angle of incidence (θ_1) increases in absolute value, the angle of reflection (θ_2) also increases to a value of 90°. There is a value of the angle of incidence $\theta_1 = \theta_{cr}$ for which the ray will reflect from the surface and not pass through it. So for $\theta_1 > \theta_{cr}$ we have a phenomenon called "Total internal

reflection" and the angle Θ_{cr} is called "critical angle." This condition will be later used in the optical fibers section.

Mathematically, for $\theta_2 = 90^o$ we get :

$$sin\theta_{cr} = \frac{n_2}{n_1} \tag{1.3}$$

1.2 Geometrical Optics

In order to analyse such a system we can either approach it by focusing on the wavefronts or by using the concept of a ray. The first option is rather difficult to manipulate mathematically, so it is convenient for us to study the propagation of rays, which is called Ray-tracing.

Cardinal Points

An optical system can be composed of several components. We can treat this system as a "Black Box" by defining the cardinal points. The cardinal points of the system are the first and second principal points, the first and second nodal points and the first and second focal points. They can be seen in Fig.1.4.

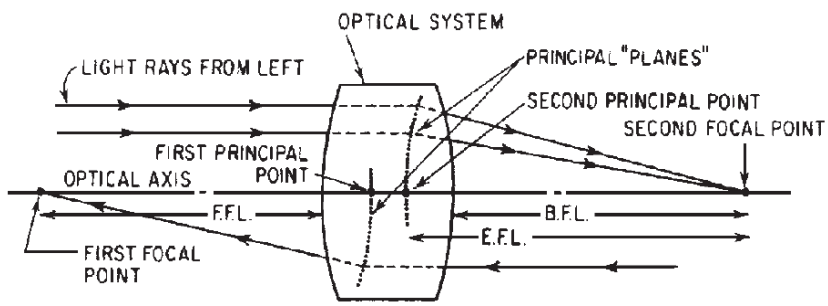


Figure 1.4: Cardinal points of an optical system[12, Page 18]

Let's assume that we have a set of parallel rays coming from an infinitely distant object. Those rays will be focused on a common point after exiting the system which is called the focus point. Now lets extent the rays entering the system and the ones exiting, The points of intersections of those rays construct a plane called the principal plane. The point of intersection of this plane with the axis is called the principal point. Rays can approach the system from either the left or the right side. The rays propagating from the left side of the system, define the second nodal point, focus point and principal point. Similarly, the ones coming from the right define the left nodal points, focus point and principal point. We also define as the effective focal length, the distance from the principal plane to the focus point. The back focal length is the distance from the back part of the lens to the focus point. Similarly we define the front focal length. Finally we need to define the nodal points of the system. The nodal points of the system are basically two axial points. If we direct a ray to intersect the first nodal point $N1$, then it appears to emerge from the second nodal point as seen in Fig.1.5. The nodal points coincide with the principal points if the system is bounded by air.

Now let's assume another system illustrated in Fig1.6a. This figure shows two pairs of conjugate rays: SD and D'F',FD and D'S'. The point D and D' lye in the principal planes at some height h and h'. We obtain the following:

$$f' = h/tan[\sigma_\rho] \tag{1.4a}$$

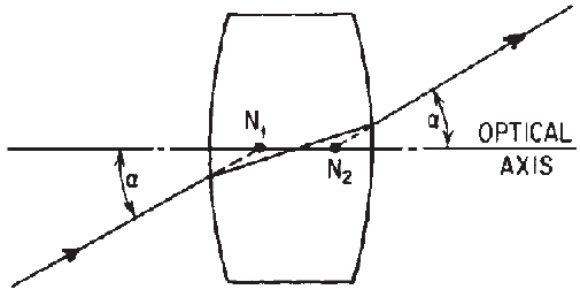
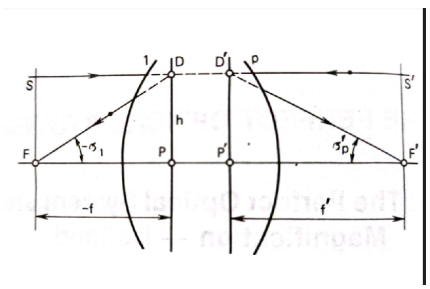
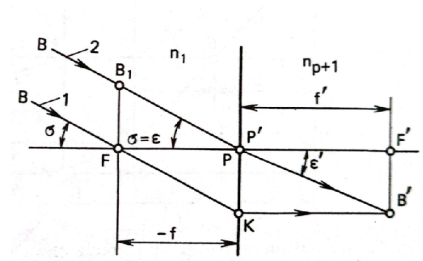


Figure 1.5: Nodal points of a system[12, Page 19]



(a) Another illustration of an optical system



(b) Passage of a ray through a perfect optical system

Figure 1.6: Two optical systems[3, Pages 32,33]

$$f = h/tan\sigma_{-1} \tag{1.4b}$$

If the height h is small , the rays travel close to the optical axis and they are called par-axial rays. The paraxial region of an optical system is a thin region, in which the angles of the rays are equal to the sines and tangents value. As a result the angle will be written as : $\theta = sin\theta = tan\theta$. This approximation can make calculations easier to manipulate and faster. Using this paraxial approximation ,the focal points can now be defined as:

$$f' = h/\sigma_\rho \tag{1.5a}$$

$$f = h/\sigma_{-1} \tag{1.5b}$$

A perfect optical system can be seen in Fig.1.6b. Both principal planes coalesce. Let's assume that we have a different refractive index n_1 and $n_{\rho+1}$. We get that FB_1 = PK=$F'B'$. That leads us to the following :

$$-ftan\epsilon = f'tan\epsilon' \tag{1.6}$$

As $\epsilon -> 0$, $\epsilon' -> 0$. Therefore, for small ϵ we have

$$-fsin\epsilon = f'sin\epsilon' \tag{1.7}$$

Using snell's law and equation Eq.1.2 we get :

$$-f'/f = n_1/n_{\rho+1} \tag{1.8}$$

From which we conclude that if the system is bounded by the same medium the first and second focal lenghts have the same value.

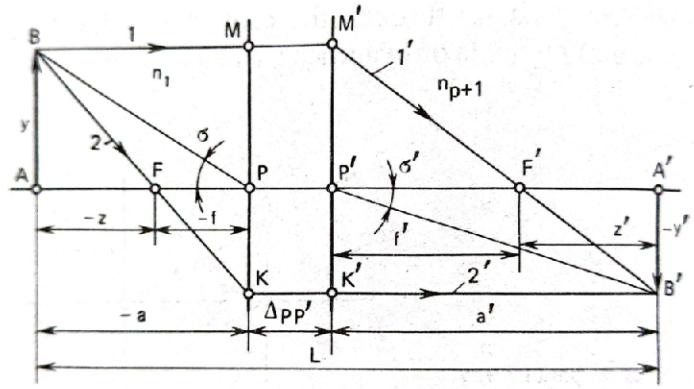


Figure 1.7: Rays coming from an object and forming an image[3, Page 34]

Object-Image relationships in Geometrical Optics by using Ray-Tracing

Before moving on, it is essential to define some sort of consistent sign convention for practical necessity.

1. Heights above the optical axis are positive.
2. Distances measured to the left of a reference point are negative and the ones to the right are positive.
3. The focal length of a converging lens is positive and the one of a diverging is negative

In Fig.1.7 a system is being illustrated by its cardinal points. The object is the line AB The image that is being created is the line A'B'. The ray labelled as 1 is travelling parallel to the optica axis until it meets the first principal plane point called M. It changes direction as it exits the second principal plane point M' and it passes through the second focus F'. The ray labelled as 2 passes through the first focus point and enters the first principal plane at K and it exits at K', propagating parallel to the optical axis. The point of intersection of these two rays is the B' point. And thus the image is being created.

Now from similar triangles (ABF and F'A'B') we get :

$$\frac{-y'}{y} = \frac{-f}{-z} = \frac{z'}{f'} \tag{1.9}$$

From the second part of the Eq.1.9 we conclude that :

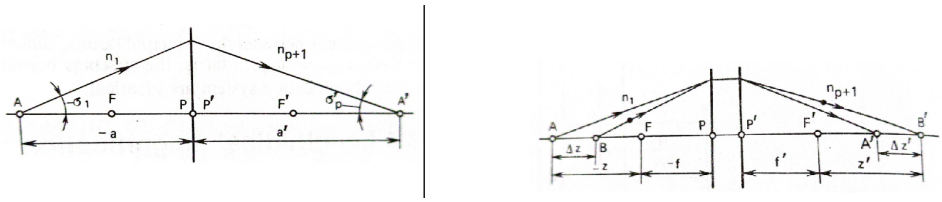
$$zz' = ff' \tag{1.10}$$

Eq.1.10 is called Newton's conjugate distance equation. For a system which is bounded by the same medium we get $f' = -f$. Eq.1.10 now becomes :

$$zz' = -f'^2 \tag{1.11}$$

We know set as $z = \alpha - f$ and $z' = \alpha' - f'$ from Fig.1.7, where α and α' are the distances from the principal planes to the image and the object. Eq.1.10 now becomes :

$$\frac{f'}{\alpha'} + \frac{f}{\alpha} = 1 \tag{1.12}$$



(a) Angular Magnification (b) Longitudinal

Figure 1.8: Two different types of magnifications[3, Pages 37,38]

If we set f=-f' Eq.1.12 takes another form:

$$\frac{1}{\alpha'} + \frac{1}{\alpha} = \frac{1}{f'} \tag{1.13}$$

Angular and Longitudinal Magnification

Firstly, we define the transverse or lateral magnification of the system as the ratio of image to object height, referring to Fig.1.7:

$$\beta = \frac{y'}{y} = \frac{-f}{z} = \frac{-z'}{f} \tag{1.14}$$

From Fig.1.7 we set $z = \alpha - f$ and $z' = \alpha' - f'$ and we get :

$$\alpha' = \frac{(\beta - 1)f}{\beta} = \frac{n_1}{n_{\rho+1}} \frac{1-\beta}{\beta} f' \tag{1.15a}$$

$$\alpha' = \frac{1-\beta}{\beta} f' \tag{1.15b}$$

The angular magnification γ is defined as the ratio of the tangents of angles the ray makes with the optical axis on the image and object side and it is given by the following equation :

$$\gamma = \frac{tan\sigma'_\rho}{tan\sigma_1} \tag{1.16}$$

If we take a perfect optical system, in which the principal planes coincide, illustrated in Fig.1.8a, we get that $\gamma = \alpha/\alpha'$.

Now, by using Eq.1.15a and Eq.1.15b we get:

$$\gamma = \frac{n_1}{n_{\rho+1}} \frac{1}{\beta} \tag{1.17}$$

The longitudinal magnification is the ratio of an infinitesimally small axial line segment in image space to the conjugate line segment in object space :

$$\alpha = \frac{dz'}{dz}; \tag{1.18}$$

In Fig.1.8a we can see the conjugate line segments $\Delta z'$ and Δz whose ratio in the limit of $\Delta z -> 0$ gives the longitudinal magnification. If we differentiate Eq.1.10 we get that $zdz' + z'dz = 0$ and thus Eq.1.8b becomes :

$$\alpha = -\frac{z'}{z} \tag{1.19}$$

Now, if we use Eq.1.10 into the previous one we get:

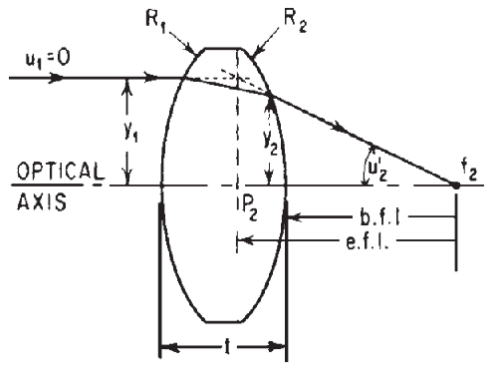


Figure 1.9: A ray propagating through a lens[12, Page 33]

$$\alpha = -\frac{ff'}{z} \tag{1.20}$$

We can also write Eq.1.20 by using the lateral or transverse magnification β in order to find a relationship between the angular and longitudinal magnification:

$$\alpha = \frac{n_{\rho+1}}{n_1}\beta^2 \tag{1.21a}$$

$$\alpha\gamma = \beta \tag{1.21b}$$

Focal Points and Principal Points of a Thick-Lens Element

In Fig.1.9,a parallel ray is illustrated passing through a lens. In this section we will attempt to calculate the effective focal length and the back focal length. We previously mentioned that the E.F.L. is measured from the principal plane and the B.F.L. is measured from the back of the lens.

Using once again the paraxial region approximation we get :

$$e.f.l = f = \frac{y_1}{u_2'} \tag{1.22a}$$

$$b.f.l = \frac{y_2}{u_2'} \tag{1.22b}$$

Before moving on, we need to define the radius of Curvature as R_1 and R_2, the thickness of the lens as t, and the power as ϕ. Using some relation equations from [12, Section 2.7] we get that:

$$\phi = \frac{1}{f} = (N-1)[\frac{1}{R_1} - \frac{1}{R_2} + \frac{t(N-1)}{R_1R_2N}] \tag{1.23a}$$

$$b.f.l. = f - \frac{ft(N-1)}{NR_1} \tag{1.23b}$$

As seen in Fig.1.9 we can calculate the distance from the second surface to the second principal plane by subtracting the b.f.l. from the e.f.l. Using the same procedure but in reverse we can calculate the front focal length and the first principal plane as well.

The position of the principal planes and focal points can vary, depending of the type of the lens and its characteristics. In Fig.1.10 we can clearly see how the position of these points change.

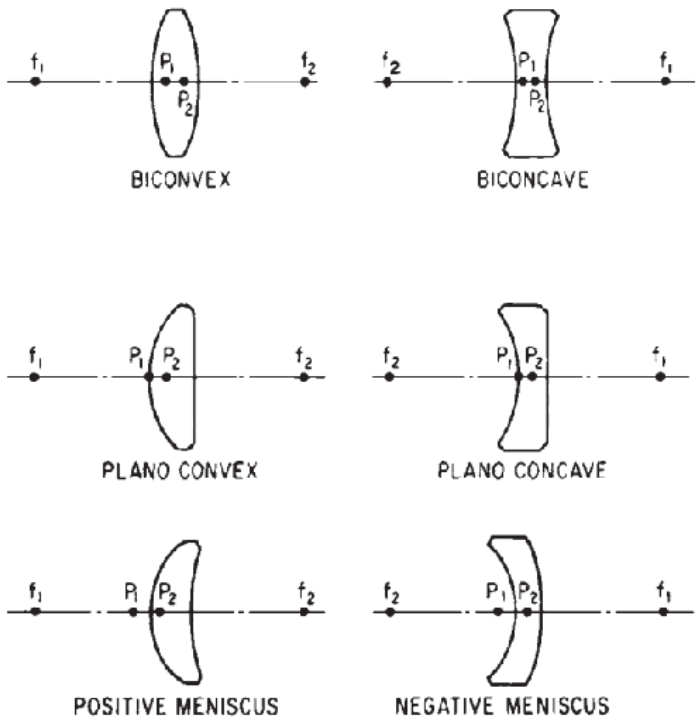


Figure 1.10: The position of the principal planes and focal points of different lenses[12, Page 36]

The Thin Lens

Now if the thickness of the lens is small enough, we can approximate the e.f.l. and the b.f.l. by assuming that $t = 0$. Equations 1.23a and 1.23b now become:

$$\frac{1}{f} = (N-1)(\frac{1}{R_1} - \frac{1}{R_2}) \tag{1.24}$$

Obviously now the principal planes of the thin lens are coincident with the location of the lens. Therefore the e.f.l. and the b.f.l. have the same value.

System of Separated Components

In this section, we will treat an optical system, composed of two lenses, in terms of the focal lengths and spacings. First lets assume that we have a system of a simple bi convex lens shown in Fig.1.11. We have alreay showned that $u = \frac{y}{s}$ and $u' = \frac{y}{s'}$.Using the previously proven equation 1.13 with $a = S$,$a' = S'$ we get :

$$u' = u + \frac{y}{f} = u + y\phi \tag{1.25}$$

We also need to use the transfer equations in [12, Section 2.6]

$$y_2 = y_1 - du_1' \tag{1.26a}$$

$$u_1' = u_2 \tag{1.26b}$$

The previous equations can be applied in a thick lens system. Now lets move on into a 2 component system illustrated in Fig.1.12. Our goal is to calculate the e.f.l. of the whole

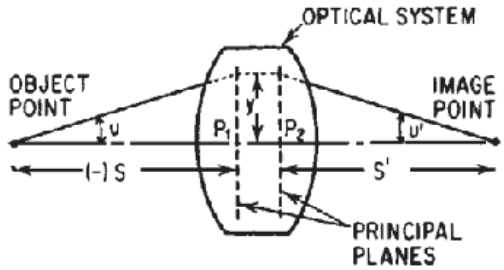


Figure 1.11: A simple bi convex lens system[12, Page 39]

system. Let's assume that we have two lenses of power ϕ_α and ϕ_β, separated by a distance d. If both lenses are thick, then d is the distance of the second principal plane of the first lens with the first principal plane of the second one as seen in Fig.1.12. Now let's a parallel ray through our system.

Beginning with the parallel ray hitting the first lens we get

$$u_\alpha = 0 \tag{1.27a}$$

$$u'_\alpha = 0 + y_a\phi_\alpha \text{ by Eq.1.25} \tag{1.27b}$$

$$y_b = y_\alpha - dy_\alpha\phi_\alpha \text{ by Eq.1.26a} \tag{1.27c}$$

$$u'_\beta = y_\alpha\phi_\alpha + y_\alpha(1 - d\phi_\alpha)\phi_\beta \text{ by Eq.1.25} \tag{1.27d}$$

$$= y_\alpha(\phi_\alpha + \phi_\beta - d\phi_\alpha\phi_\beta)$$

The total power of system $\phi_{\alpha\beta}$ is defined as:

$$\phi_{\alpha\beta} = \frac{1}{f_{\alpha\beta}} = \frac{u'_\beta}{y_\alpha} = \phi_\alpha + \phi_\beta - d\phi_\alpha\phi_\beta = \frac{1}{f_\alpha} + \frac{1}{f_\beta} - \frac{d}{f_\alpha f_\beta} \tag{1.28}$$

Now we solve for $f_{\alpha\beta}$:

$$f_{\alpha\beta} = \frac{f_\alpha f_\beta}{f_\alpha + f_\beta - d} \tag{1.29}$$

The b.f.l. is given by:

$$b.f.l. = \frac{y_b}{u'_b} = \frac{f_{\alpha\beta}(f_\alpha - d)}{f_\alpha} \tag{1.30}$$

In order to calculate the f.f.l. we reverse the procedure and we send the ray from the right side of the system. Similarly we end up with:

$$(-)f.f.l == \frac{y_b}{u'_b} = \frac{f_{\alpha\beta}(f_\beta - d)}{f_\beta} \tag{1.31}$$

Before moving on, we can calculate the new principal planes of the new 2 lens system, by subtracting the b.f.l. from the e.f.l. and similarly the f.f.l. from the e.f.l.

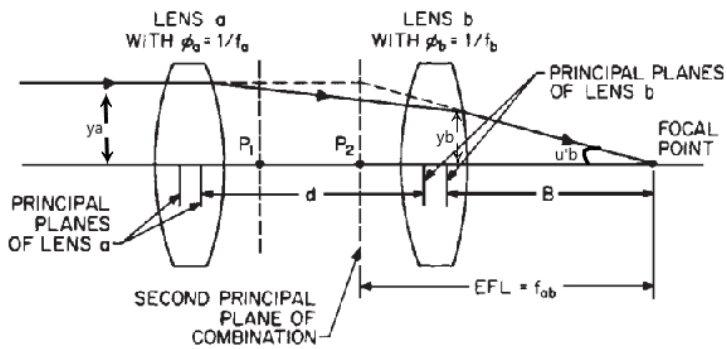


Figure 1.12: A parallel ray passing through two separated lenses[12, Page 41]

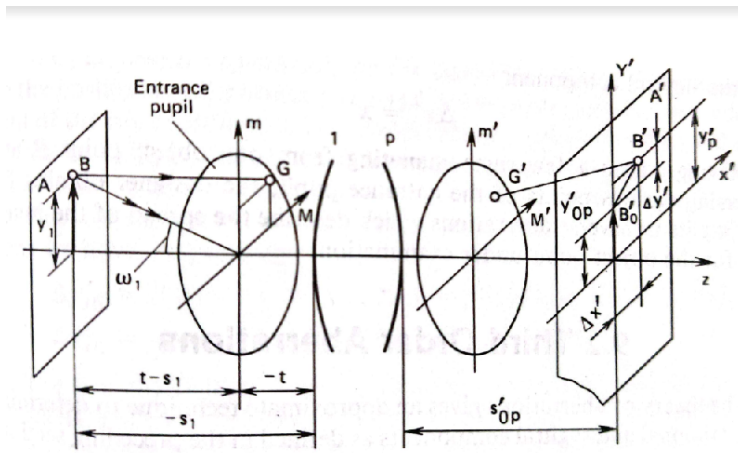


Figure 1.13: Aberrations of a skew ray[3, Page 155]

1.3 Aberrations of a system

In the previous sections, we analyzed systems by focusing on the paraxial region. Now, we will study the behavior of real systems with finite apertures and fields of views. An ideal system or similarly a well-corrected system behave according to the rules of the paraxial region. Real systems(or not corrected ones) are not able to focus light at one point beyond the system. These violations are called aberrations of a system. In other words, we will measure the amount by which rays miss the paraxial image point. There are two types of aberrations:

1. Monochromatic aberrations which are aberrations created by the propagation of monochromatic light through an optical system.

2. Chromatic aberrations that have to do with the variation of the wavelength of light propagating through the system.

Both aberrations can occur in a real system, but in this report we will focus on monochromatic aberrations only.

Let us begin by tracing a skew ray BG in Fig.1.13. The object point B is elevated at a distance y_1, t is the distance from the front surface to the entrance pupil,s_1 is the distance between the front surface and the object plane A. If the coordinates y_1,t,s_1 are known, then the position the ray can be known by specifying the point G where it pierces the entrance

pupil, namely m in the y axis and M in the x axis. Using the previous parameters [3, Section 8.2] we can derive the coordinates of point $B'(y', x')$ at which the ray pierces the image plane. The distance S'_0p can also be derived by tracing a quasi-paraxial ray. This trace leads to the size of the ideal image :

$$y'_{0p} = y_1\beta_0 \tag{1.32}$$

with β_0 being the transverse magnification of a perfect system. Taking another look at Fig.1.13 the transverse aberration is characterized by the line segment $B'B'_0$ which is represented in terms of the axes of the image plane:

$$\Delta y' = y'_p - y'_{0p} \text{ Meridional component} \tag{1.33a}$$

$$\Delta x' = x' \text{ Sagittal component} \tag{1.33b}$$

Third-Order Aberrations

The theory of aberrations aims to determine these meridional and sagittal components. The components $\Delta x'$ and $\Delta y'$ are functions of the ray coordinates $y_1(\omega_1), m, M$. The theory finds a relation between those parameters and the components $\Delta x', \Delta y'$ as:

$$\Delta y' = f(y_1, m, M) \tag{1.34a}$$

$$\Delta x' = f(y_1, m, M) \tag{1.34b}$$

Because the system is symmetrical about the optical axis, the equations 1.34a,1.34b contain no terms of ever order. We can now develop these equations into a power series wiht only odd order terms:

$$\Delta y' = \Delta y'_{III} + \Delta y'_V + \Delta y'_{VII} + ... \tag{1.35a}$$

$$\Delta x' = \Delta x'_{III} + \Delta x'_V + \Delta x'_{VII} + ... \tag{1.35b}$$

The terms appearing on the right side of these equations are called respectively third-order, fifth-order, etc. meridional and sagittal components of aberrations. The third order ones are called primary aberrations and the fifth order one are called secondary aberrations. The other higher order components are called higher-order aberrations. Aberrations described by higher orders are extremely cumbersome and inconvenient and therefore we will focus on the third order ones.

Spherical Aberration

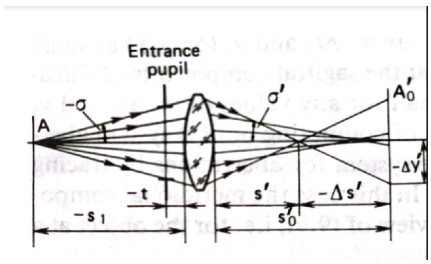
Consider an optical system with a single bi-convex lens illustrated in Fig.1.14a. The position of the ideal image A'_0 of the object point A is located by a paraxial ray which intersects the optical axis at a distance of s'_0 from the back of the lens. As the ray height increases, the position of the ray intersection with the optical axis moves further back from the paraxial focus point. There are two types of spherical aberrations:

1. Longitudinal Spherical aberration: The distance from the paraxial focus to the axial intersection of the ray.

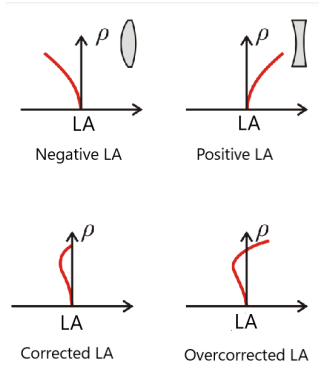
$$\Delta s' = s' - s'_0 \tag{1.36}$$

2. Transverse Spherical aberration : The aberration measured to the "vertical" direction.

$$\Delta y' = \Delta s' tan\sigma' \tag{1.37}$$

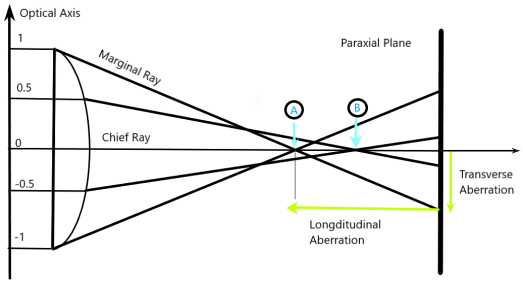


(a) System with spherical Aberration[3, Page 162]



(b) Plots of longitudinal and transverse aberrations of different systems

Figure 1.14: Analysing the spherical aberration



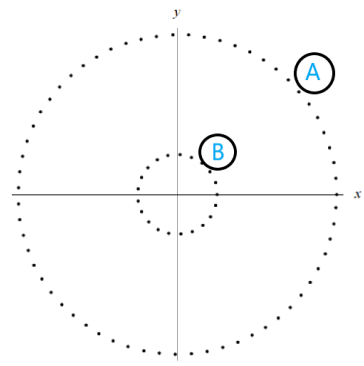


Figure 1.15: Illustration of the Spherical Aberration of a system thought the marginal rays.

Using the sign convention, spherical aberration with a negative sign is called undercorrected spherical, associated with positive lenses. Similarly, positive spherical is called overcorrected because it is associated with diverging elements. The spherical aberration of a system usually represented graphically. Such plots can be seen in Fig.1.14b in which the first row of the plots represent the longitudinal and the second row the transverse aberrations of the systems.

To further analyse and visualize the spherical aberration on a system we can use the marginal, half marginal rays and the chief ray[9]. Marginal ray is a term used to describe one of the rays of light that passes through the optical system, specifically the ray that travels through the outermost edge or margin of the optical aperture. The marginal rays and the spot size of these rays can be seen in Fig.1.15.

How to minimize the Spherical Aberrations

There are several ways to reduce the spherical aberration of a system:

- **The Stop Method**: The spherical aberration of a lens strongly depends on the aperture of the lens. In other words, the rays that pass closer to the edge of the lens produce greater aberration values. We can eliminate these marginal rays , by using a stop, and thus allowing only rays that are closer to the paraxial region. A demonstration of this technique can be seen in Fig.1.16. A drawback of this method is that it reduces the light passing through the lens.

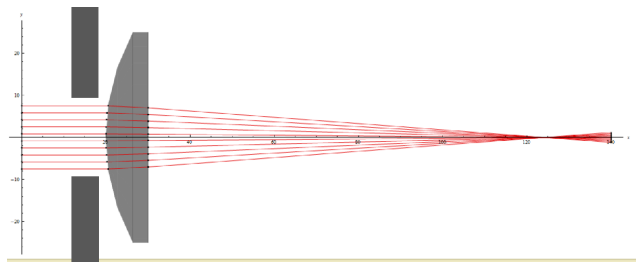


Figure 1.16: Stop Method

- **Aspheric Lenses**: Aspheric lenses can be designed in such a way that they can converge light into one point. Such a lens can be seen in Fig.1.17.

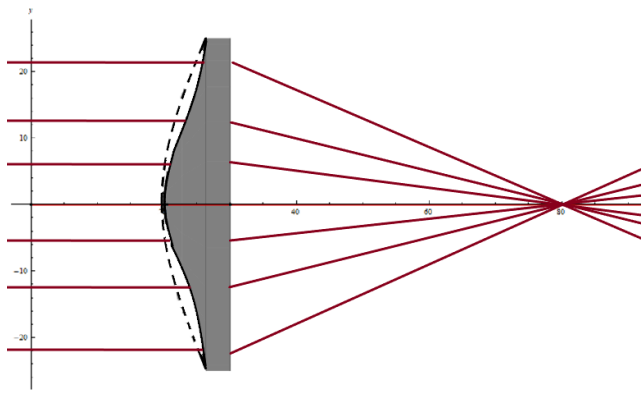


Figure 1.17: Simplified Aspheric Lens design.

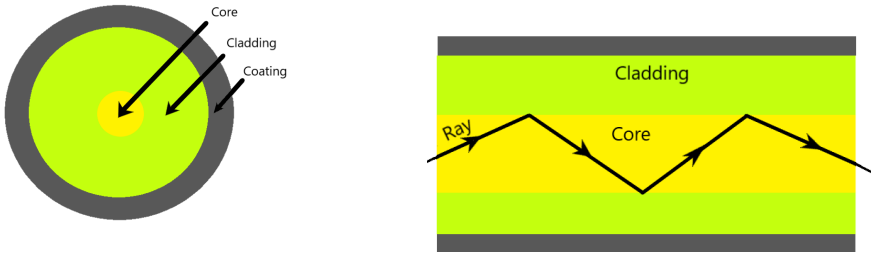
- **Lens Splitting**: Two converging lenses can be used instead of one. That will greatly reduce the spherical aberration. Those two lenses must have the same optical power when combined, compared to the lens they have replaced. Such a system can be seen in Fig.1.12.
- **Crossed lens design**: A spherical lens with the facing radius being the 1/6[5] of the other radius of curvature.Such a design will be later used in Fig.3.8.
- **Gradient Index Materials**:Lenses with varying refractive indices across their surfaces[8].
- **Combination of convex and concave lens**: A convex lens has negative aberration while a concave one has positive aberration. By combing these two lenses, their aberrations cancel out. This technique will be later used in Fig.3.14.

1.4 Optical Fibers

General

Optical Fibers have changed the world from the moment they were firstly used. Despite their diameter size, which is similar to the human hair one, they are very durable and can be rather long. They are mainly made of glass and plastic and light is being propagated through them.

Compared to other optical systems using metallic conductors, optical fiber systems have many advantages such as: electrical isolation, freedom from electromagnetic interference,



(a) The structure of an optical fiber. (b) A ray propagating through the fiber.

Figure 1.18: Optical Fiber analysis.

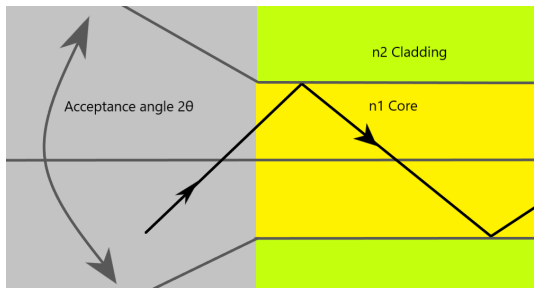


Figure 1.19: Numerical Aperture of a fiber.

low power loss and they are lighter and smaller as well. Some of the fiber optic systems applications are : Communications, Sensing, Power delivery and illumination .

Fiber construction

An Optical fiber consists of 3 main components as seen in Fig.1.18a.: The core, the cladding and the coating[13].

1. Core :The material of the core part is usually made of plastic or glass. The core of the optical fiber is the part where the light is being propagated.
2. Cladding :The cladding part of the fiber is typically made of the same material as the core. The only difference is the lower refractive index of the material used in the cladding compared to the core.
3. Coating : The coating of a fiber usually consists of one or more coats of plastic material to protect the fiber from the physical environment. Several types of material can be used depending on the usage of the optical fiber.

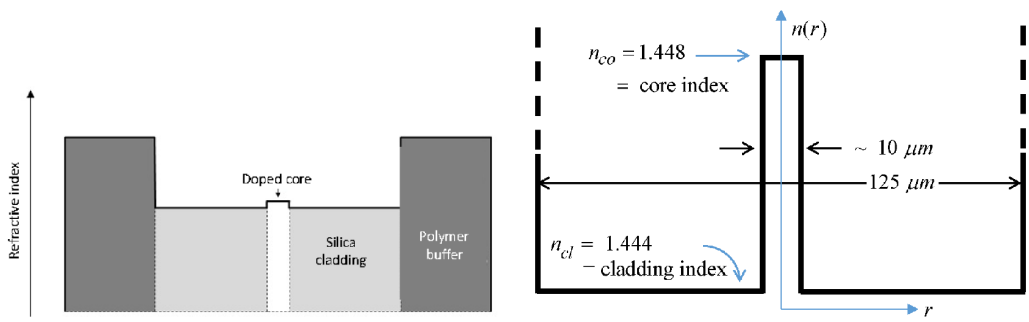
Numerical Aperture

The numerical aperture (NA) of an optical fiber is defined as the measure of maximum angles at which light rays can enter the fiber. A graphical representation of the N.A. can be seen in Fig.1.19.The NA concept will analyzed later on. For now let us defined it as :

$$NA = \sqrt{n_{core}^2 - n_{cladding}^2} = sin\theta \tag{1.38}$$

Fiber Analysis

A simple diagram of the index profile of each part of the fiber can be seen in Fig.1.20a. This is called a step-index optical fiber. The difference of the refractive index(Eq:1.2) is what enables the ray to travel across the fiber, according to the total refraction condition(Eq.1.3) if Snell's Law. Such a reflection can be seen in Fig.1.18b. We can see that the light only propagates in the core area and not outside of it. Finally there is a certain way we specify a fiber: Diameter of core, cladding and the coat. So a 60/120/250 fiber would refer to a fiber with a core diameter of 60 μm, a cladding of 120 μm and a coating of 250 μm[6].



(a) Index profile of each element of the fiber.

(b) RIP for a step-index profile.

Figure 1.20: Examples of the refractive indexes of fibers[4]

A fiber system can be analysed using two approaches. The first is by using ray tracing technique and the other one is called electromagnetic of modal analysis. The second one uses the solutions from the Maxwell's equations. Those solutions are called modes. A simple example of these two different ways of approaching this kind of system, can be seen in Fig.1.21. Now let's talk about these solutions called modes. Let's start by defining the

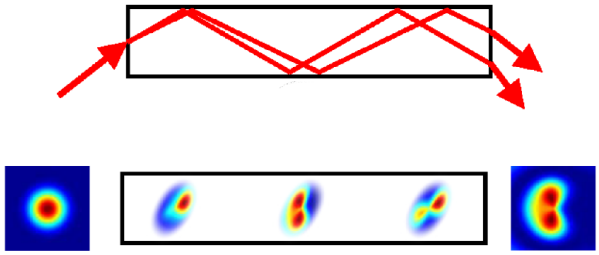


Figure 1.21: Two seperate ways of fiber analysis[4]

propagating field E. The idea is to solve the Maxwell's equations in order to understand how the light is being propagating through the fiber. The field E will dependant from the x,y,z axis and the time as well.The light propagates in the Z-direction as seen in Fig.1.22. By solving the MaxWell's equations, with the appropriate boundary conditions we can find the analytically the field E as the general solution of these equations:

$$E(x, y, z, t) = E_0(x, y)cos(\beta z - \omega t) \tag{1.39}$$

where

- $E_0(x.y)$ = field amplitude distribution in the transverse plane.
- $\omega = 2\pi f$ where f is the optical frequency.

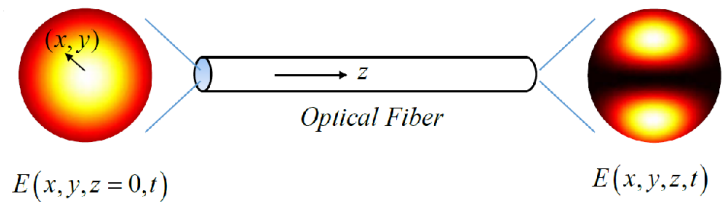


Figure 1.22: Fluctuation of the field E as the light propagates through the fiber[4]

- $\beta = 2\pi n_{eff}/\lambda$ = propagation constant
- n_{eff}=an effective refractive index
- λ = optical wavelenght

Writting the Eq.1.39 in that form, we have made some assumptions. The light is monochromatic with a frequency f. The fiber refractive index depends only on the transverse coordinates (x,y) and not on z. The electric field is a vector quantity so the solution will be of the form $E = E_x\hat{x} + E_y\hat{y} + E_z\hat{z}$. Finally the fiber is lossless, because the E_0 is only dependant of x,y.

One key element of a fiber is the transverse spatial variation of the refractive index n(x,y). We mentioned that is not dependant of the z and it also has circular symmetry. In other words, its value depends only from the radial distance $r = \sqrt{x^2 = y^2}$. The dependence of the refractive index by r is called the "refractive index profile" or "RIP". By knowing the RIP we can lock all of Maxwell's equations solutions. The RIP profile of a fiver can be seen in Fig.1.20b. Another parameter specified is the delta parameter and it is defined as :

$$\Delta = \frac{n_{co}^2 - n_{cl}^2}{2n_{co}^2} \tag{1.40}$$

For example, the optical fibers used in communications are weakly-guiding fibers in which $n_{co} \approx n_{cl}$. This gives a Delta parameter value of $\Delta << 1$.

Modes in step-index Fibers

Mathematically, fiber modes are eigenfunctions of the system defined by MEs and the RIP. There are three mode categories allowed in an optical fiber:

1. Core modes and modes guided by the core-cladding interface. They are finite in number and discrete.
2. Cladding modes are modes guided by the combination of the core and cladding and also the combination of a cladding with a buffer-coat interface.
3. Radiation modes which are unguided waves that are valid solutions but not confined by the fiber. They is an infinite amount of them.

Each mode is identified by a label of l and m. We will write Eq.1.39 in a complex form, in order to visualize these labels.

$$E_{lm}(x, y, z.t) = e^{j(\beta_{lm}z - \omega t)}E_{0lm}(x, y) \tag{1.41}$$

Some important properties are represented below:

- From Eq.1.41 we get that each mode is characterized by a specific field distribution $E_{0lml}(x, y)$ and a constant $\beta_{lm} = n_{eff,lm}(\frac{2\pi}{\lambda})$.

- The value of the transverse field distribution $E_{0lm}(x, y)$ remains the same in the Z direction. The only change is the phase part of Eq.1.41.

- Similarly, the field distribution $E_{0lm}(x, y)$ is independant of the direction of propagation. For waves propagating in the Z direction the argument is $(\beta_{lm}z - \omega t)$ and in the -Z direction its $(\beta_{lm}z + \omega t)$.

- We mentioned that modes are mathematical eigenfunctions that are orthogonal and form a basis. As a result, the field distribution can be expressed as linear combination of the modal solutions:

$$E_{in}(x, y, z) = \sum_{lm} c_{lm} E_{0lm}(x, y) + \int_0^\infty \int_0^\infty c_{\mu\nu} E_{0\mu\nu}(x, y) d\mu d\nu \tag{1.42}$$

  where $E_{in}(x, y)$ is the transverse distribution of an arbitrary light beam incident on the fiber endface. The c_{lm} are complex weighting coefficients where the discrete indices l and m are guided modes and the continuous indices μ and ν label the radiation modes. In real life, most of the coefficients are zero or close to zero because typically not all the modes are needed to represent an incident field. In other words, if we inject a light beam into a fiber, when it exits it, some modes will be excited, with their coefficients not being zero.

- These modes are orthogonal to each other, given that they are eigenfunctions with orthogonal bases. As a result, there is never intermodal crosstalk. Light in one mode stays in that mode and it does not interfere with other modes. In case of any bents, deformations or density fluctuations, the assumption of the invariance of the index along z is invalid and it canl lead to some crosstalks.

- Some modes can have exactly the same propagation constant even though they have different l and m labels. These modes are called degenerate and they can exchange power even in the absence of perturbations.

The complex coefficients c_{lm} are a normalized measure of how well any particular mode E_{0lm} mathes the incoming field distibution $E_{in}(x, y, z = 0)$. We get that :

$$|c_{lm}|^2 = \frac{\left|\int\int E_{in}(x, y, z = 0) E_{0lm}(x, y) dxdy\right|^2}{\int\int |E_{in}(x, y, z = 0)|^2 dxdy \int\int |E_{0lm}(x, y)|^2 dxdy} \tag{1.43}$$

The integrals are performed over the entire transverse plane. The c_{lm} are proportional to the overlap integral used in traditional optics. In other words:

$$c_{lm} \propto \int\int E_{in}(x, y, z = 0) E_{0lm}(x, y) dxdy \tag{1.44}$$

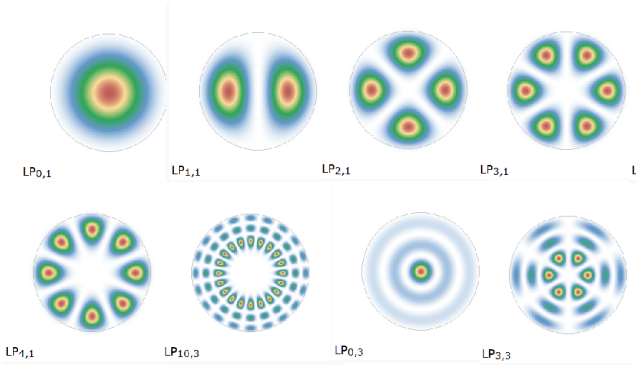
This is the inner product of the input field and the mode field and it is also the projection of the input field onto the mode. To conclude, using the previous analysis we get those 3 important properties:

1. The modes form a basis so any arbitrary incident field can be expressed as a linear combination of modes.

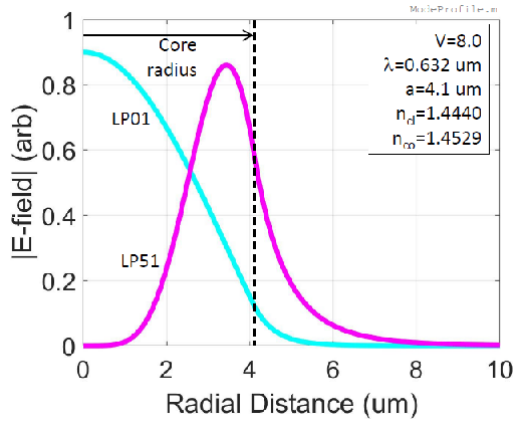
2. Each mode propagates along the z axis, without interfering with other modes.

3. To obtain the total field contribution, we simply propagate each mode separately and then we add them all up.

Now, we are ready to visualize and understand the shape and parameters of each mode.



(a) The transverse field distributions $LP_{lm}(x, y)$. The difference in colors represents the phase difference between the fields[11].



(b) The mode profiles of LP_{01} and LP_{51}[4].

Figure 1.23: Visualization of the fiber modes

Core modes in Step-Index Fibers

LP modes are the modes that are linearly polarized, meaning that the transverse field contribution ends almost completely in either E_x or E_y. The field distributions for some low LP modes are shown in Fig.1.23a. LP modes are designated by the number l and m that we mentioned previously. Even though the boundary conditions imposed by the RIP have perfect circular symmetry, we see that only the l=0 modes are circularly symmetric. The other LP modes have rotational symmetry , under rotations of $2\pi/l$ or π/l. Another important notice is that the different LP modes are not entirely confined with in the core. Some higher LP modes have more light near or beyond the core-cladding interface, compared to the lower order ones. Such a behavior can be seen in Fig.1.23b. It is clear that the higher order LP_{51} mode propagates the light near to the edge of the core.

Now, let us examine some of the different behaviors of the LP modes in a step-index fiber. We mentioned that the phase of the field distributions of the modes change according to the following equation:

$$LP_{lm}(x, y, z) = e^{j\beta_{lm}z} LP_{0lm}(x, y, z) \tag{1.45}$$

where the phase accumulates at a constant rate β_{lm} per meter. The core with the biggest β_{lm} constant (largest refractive index) is called the Fundamental mode. This mode accumulates phase at the largest rate according to Eq.1.45 and it has the lowest phase speed acoording to Eq.1.46.

We now define the phase speed of each mode as:

$$u_{phase} = \frac{\omega}{\beta} = \frac{c}{n_{eff}} \tag{1.46}$$

where $n_{eff} = \beta/k$ and $k = 2\pi/\lambda$ which is the wavenumber in vacuum.

The speed at which power flows is given by the group speed which is defined as:

$$u_{group} = (\frac{d\beta}{d\omega})^{-1} = \frac{c}{n_{group}} = \frac{c}{(n_{eff} + \omega(dn_{eff}/d\omega))} \tag{1.47}$$

Single- vs Multi-Mode Fiber

When solving the MEs for the core-guided modes in a step-index fiber , we come across a usefull parameter which is called the "V-parameter" or the "normalized frequency" and it

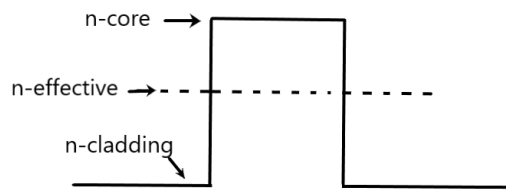


Figure 1.24: The n_{eff} as a value.

is given by:

$$V = \frac{2\pi}{\lambda}\alpha\sqrt{n_{co}^2 - n_{cl}^2} = \frac{2\pi}{\lambda}\alpha NA = \omega\frac{\alpha}{c}\sqrt{n_{co}^2 + n_{cl}^2} \tag{1.48}$$

where α is the radius of the core. For a given optical frequency and RIP, the value of V determines the number of core-guided modes the fiber can support. In other words, there is a value for V called V_{cutoff}=2.405 for which the mode is either in single mode or in multi mode.

- If $V \leq V_{cutoff} = 2.405$, there is one core-guided solution and the fiber is called singlemode.
- If $V > V_{cutoff} = 2.405$ more than one core-guided mode is allowed in the fiber. That fiber is called multi-mode fiber.

Before moving on, we need to mention the following:

1. The V parameter is dimensionless and depends on the difference between the core and the cladding indices and on the ratio of the core radius to the optical wavelength.
2. The terms "Single mode" and "Multi mode" apply only to core-guided modes.
3. A single mode fiber can still have many allowed cladding and radiation modes.
4. Each core is characterized by an effective refractive index $n_{eff} = \beta/k$ that lies somewhere between the cladding and core indices shown in Fig/1.24. Analytically we have:

$$n_{cladding} \leq n_{effective} \leq n_{core} \tag{1.49}$$

The exact value of n_{eff} is determined by a parameter b called normalized propagation constant. Therefore we have:

$$n_{eff}^2 = (1-b)n_{cl}^2 + bn_{co}^2 \tag{1.50}$$

For a weakly guided approximation we have :

$$n_{eff} \approx (1-b)n_{cl} + bn_{co} \tag{1.51}$$

Solving for b we get:

$$b = \frac{(\beta/k)^2 - n_{cl}^2}{n_{co}^2 - ncl^2} \approx \frac{(\beta/k) - n_{cl}}{n_{co} - c_{cl}} = \frac{n_{eff} - n_{cl}}{n_{co} - n_{cl}} \tag{1.52}$$

In order to understand the LP modes better, we can plot the propagation constant as a function of the V-number, shown in Fig.1.25.

By analysing the plot we get that:

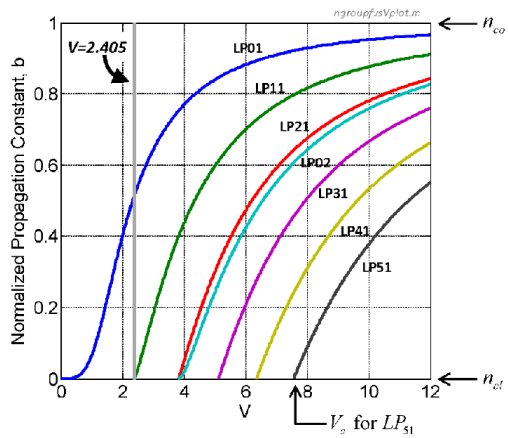


Figure 1.25: The propagation constant as a function of V[4]

1. For a mode to be core-guided, its refractive index must be lie between the core and cladding indices. In other words, $0 \leq b \leq 1$ which is the same as $n_{cl} \leq n_{eff} \leq n_{co}$.

2. While the value of V decreases, there is a point for every mode where $n_{eff} = n_{cl}$. At this point, the wave can not distinguish its own effective index from the cladding one. As a result, it is no longer core-guided. The wave has reached its cut-off point and that mode is not longer a solution of the MEs. In Fig.1.25 the mode LP_{51} reaches cut-off at $V \approx 7.6$.

3. For the fundamental mode, b reaches zero asymptotically only as V approaches zero.

4. For $V \leq 2.405$ only the fundamental mode exists, because the other modes have reached their cut-off points. Therefore, the fiber is in singlemode.

5. As $V- > \infty$ the effective index of all mode reaches the core one, making the fiber highly mutlimode, because all the modes are far from their cut-off region.

Numerical Aperture of Optical Fibers

We previously defined the NA in Eq.1.38. That is the correct definition for a step-index profile. For other types of fibers with different profiles, the concept of the NA becomes questionable. In that case, the maximum input ray angle will depend on the position of the input surface.

- For a single mode fiber, the NA is typically of the order of 0.1, but they can vary roughly between 0.05 and 0.4.
- For a multi-mode fiber, the NA can take values from 0.3 and larger.

A higher NA value can have the following affects:

1. Stronger beam guidance. That can be derived by the "V-parameter" defined in Eq.1.48.

2. Single mode guidance requires a small core diameter. The divergence of the beam exiting is high and the corresponding area is smaller. Thus a large mode area sinle mode fiber must have a small NA.

3. Reduction of the influence of random refractive index variations.

4. Reduction of bend losses inside the core.

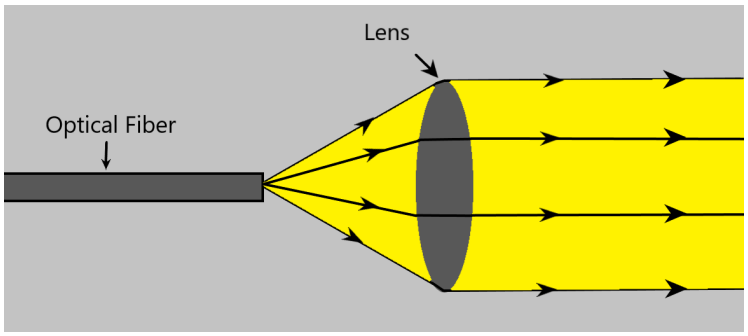


Figure 1.26: Placing the lens in the correct distance the output can be a collimated beam.

1.5 Fiber Couplers

Introduction

Fiber couplers are a broad category of optical components used in optical systems to control the distribution of light within or between optical fibers. Common types of fiber couplers include (More can be read in[1]):

1. **Optical Splitters**: Such a device splits an incoming signal into multiple output signals, depending on the use , enabling data to be sent into multiple destinations simultaneously.
2. **Wavelength division multiplexers (WDM)**: Such a coupler is used to combined or separate different signals at different wavelengths.
3. **Fiber Optics Switches** : These switches allow the routing of optical signals between different optical fibers or fiber paths.
4. **Fiber Collimators**: Fiber collimatos are used for efficient coupling of light into and out of optical fibers, including connectors, adapters, and collimating lenses.

A fiber collimator[10] is an optical device that is used to transform divergent light from an optical fiber into a parallel, collimated beam of light. Fiber collimators can be crucial components of an optical system. A simple fiber collimator can be seen in Fig.1.26. The fiber end must be placed at a distance from the lens system equal the focal length. There are two types of collimators:

- Fiber collimators attached to bare fibers. Such a design is cheap to construct but the obvious disadvantage is that the coupler is permanently attached to the fiber.
- Couplers that have a mechanical interface to a fiber connector. Some examples are FC or SMA. Some FC or SMA systems can be found here.

Instead of collimating a beam coming from a fiber, these systems can also do the opposite. To launch light from a collimated beam into a fiber or to fiber to fiber coupling.

NA of a fiber coupler

The numerical aperture of an optical coupler is given from the following expression according to Fig.1.27:

$$\mathrm{NA} = \sin\theta = \sin[\arctan\frac{D}{2f}] \approx \frac{D}{2f} \text{ for small angles} \tag{1.53}$$

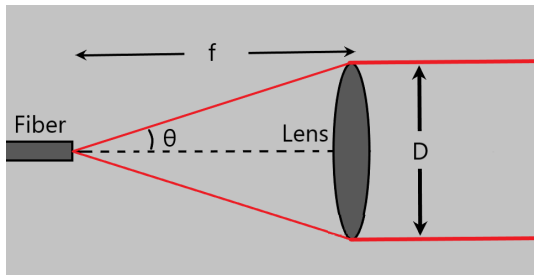


Figure 1.27: The NA of a Fiber coupler system.

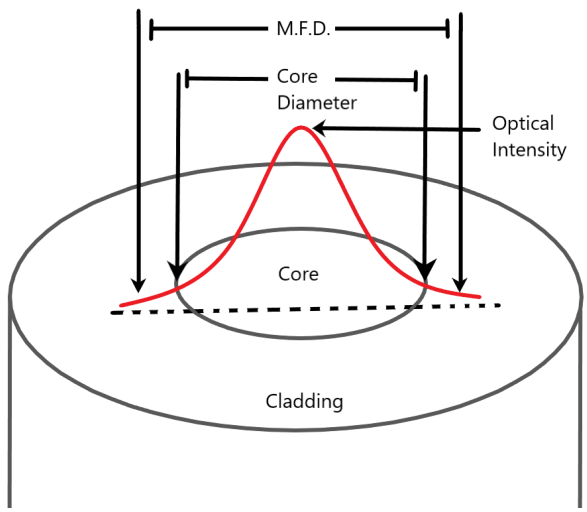


Figure 1.28: The standard intensity profile of the beam propagated through the fiber with the MFD and core diameter.

Size of the collimated beam

The beam diameter of the collimated beam can vary according to the system. Some beams can have the same diameter as the fiber diameter and in other cases, some beams can have a diameter of several millimeters or more. In a single mode optical fiber the diameter of the collimated beam can be calculated with :

$$d = 4\lambda \frac{f}{\pi[MFD]} \tag{1.54}$$

where λ is the operating wavelength of the fiber and [MFD] is the mode field diameter.

The M.F.D. is a certain measure of the beam width of light propagating through the single mode fiber. The light is mostly propagated inside the core but a small fraction propagates in the cladding as well. We can approximate the light inside the fiber a Gaussian beam, in which the M.F.D. is the diameter at which the optical power is reduced to the $1/e^2$ from its peak level. A simple illustration of the MFD can be seen in Fig.1.28

The distance between the lens and the fiber is approximately the focal length. If the lens is too close, then the output beam will diverge and if the lens is too far, then the beam will converge. From Eq.1.54, the smaller the MFD is, the larger the output beam diameter is.

We can also define the angle of divergence, which is the angle at which the beam is diverging along the optical axis:

$$\theta = \frac{[MFD]}{f}\frac{180}{\pi} \tag{1.55}$$

Note that MFD and f have the same units in this equation. Combining Eq.1.54 with Eq.1.53 we get:

$$MFD = \frac{2\lambda}{\pi NA} \tag{1.56}$$

Chapter 2

Optica Software and tools used

2.1 Intoduction to Optica Software

Optica software is a powerful and useful tool designed to optimize, streamline and revolutionize various aspects of optical design, analysis and simulation. This tool can be used by different kind of people such as students or even professional engineers and researchers.
Optica is a language extension of Mathematica used for raytracing and rendering optical system. This software enables the user to craft optical systems, simulate light propagation through intricate components and analyze the behavior of various optical elements.

2.2 Start guide and user interface

Once the user has launched Mathematica, he needs to load the Optica Software. That can be done by the following expression:

```
Needs["Optica`Optica"]
```

Now the user is ready to move on. First we need to define our system. The system usually consists of some source of rays and some components. The user has a variety of rays and components that he can use in order to create his own system. Once the system is defined, it can be propagated and then shown for the user. Before moving on, will we mention some important Build-in Optica Functions that will be used later on. The whole function list can be read in the user guide in [2]:

Firstly let us introduce some simple Ray functions:

- **LineOfRays**[linewidth, options] creates a set of rays, starting at the y axis, lying in the horizontal plane, equally distributed within the specified linewidth, and directed down the positive x axis.
- **CircleOfRays**[radius, options] creates a set of rays, starting in the y-z plane, equally distributed on the surface of a cylinder placed symmetrically about the positive x axis.
- **ConeOfRays**[conicangle, options] creates a set of rays, starting at the origin of the coordinate system, equally distributed on the surface of a cone placed symmetrically about the positive x axis.

Some build-in components functions:

- **BiConvexLens**[focallength, aperture, thickness, options] designates a lens with two equally convex spherical surfaces

- **PlanoConvexLens**[focallength, aperture, thickness, options] denotes a lens with a planar surface on one side and a convex spherical surface on the other side.
- **SphericalLens**[r1, r2, aperture, thickness, objectlabel, options] denotes a lens having spherical surfaces and a user-named objectlabel.

Some useful functions that allow us to place and change the orientation of these components or rays:

- **Move**[objectset, x, y, rotationangle, options] is used to move the relative position and orientation of a set of components and rays within a horizontal plane.
- **Move3D**[objectset, xpos, ypos, zpos, tiltvector, twistangle, options] is used to move the relative position and orientation of a set of components and rays in three-dimensional space.

Finally in order to visualize the whole system we have to use the follwing command:

- **ShowSystem**[objectset, options] takes an object set containing Ray, Component, and OpticalSystem objects and renders them according to PlotType, RayChoice, ShowRange, and ColorView options.

Now, we are ready to create our own optical systems. A simple example can be shown in the following code lines:

```
    SimpleSystem = PropagateSystem[{
    Move[LineOfRays[50, NumberOfRays -> 20], {0, 0, 0}],
    Move[PlanoConvexLens[100, 50, 10], {100.0, 0, 0}],
    Boundary[300]
    }];
```

We are using Propagate system in order to define and propagate our system at the same time. The Boundary[boundary parameters]designates a rectangular box that absorbs rays intercepted by its walls. We have set as a boundary, the value of 300 mm. We have defined set of rays that form a line, with a linewidth of 50 mm and a ray number of 20. The lens is a plano convex lens with a focal length of 100, and aperture of 50 mm and a thickness value of 10 mm that is moved in x=100 mm. Using the following command we can get an illustration of our system in Fig.2.1:

```
    XShowSystem[SimpleSystem, PlotType -> TopView, Axes -> True
        , Boxed -> False, AxesLabel -> {x, y}]
```

The user can change various parameters inside the show system command such as the type of the plot, the axes labels, the size of the figure etc. Each ray propagated has its own properties. That can be shown, be using the following command $-$ > Options[Ray]. We get the following output:

```
    Options[Ray]: {BirthPoint -> {0., 0., 0.},
        ComponentIncrement -> Automatic,
 ComponentNumber -> ComponentNumber, ConfinedNumber ->
    ConfinedNumber,
  ConfinedPosition -> ConfinedPosition,
 DiffractionMismatch -> DiffractionMismatch,
 DiffractionOrderNumber -> DiffractionOrderNumber,
 GenerationNumber -> GenerationNumber, Intensity -> 100.,
 InternalDirectionChange -> False,
 IntersectionNumber -> IntersectionNumber, IntrinsicMedium ->
    Air,
```

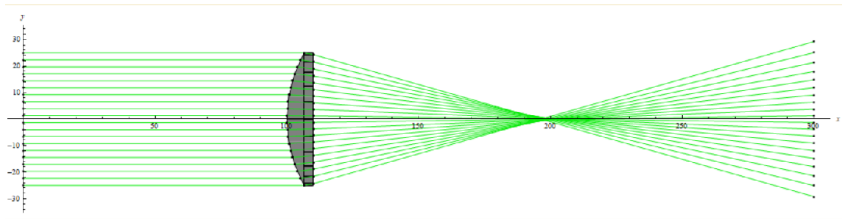


Figure 2.1: A plano convex lens system.

```
NewAuthorizedOptions -> NewAuthorizedOptions, OffAxis ->
    OffAxis,
OpticalLength -> 0., OpticalMedium -> Automatic,
Polarization -> {0., 1., 0.}, RayEnd -> {0., 0., 0.},
RayLabel -> RayLabel, RayLength -> 0., RayLineRGB -> Automatic
    ,
RayLineStyle -> {}, RayLineThickness -> 0.5,
RayPointRGB -> {0., 0., 0.}, RayPointSize -> 2., RayPointStyle
     -> {},
 RayStart -> {0., 0., 0.}, RayTilt -> {1., 0., 0.},
RefractiveIndex -> Automatic,
RotationMatrix -> {{1., 0., 0.}, {0., 1., 0.}, {0., 0., 1.}},
SurfaceBoundary -> SurfaceBoundary,
SurfaceCoordinates -> SurfaceCoordinates, SurfaceIncrement ->
    1,
SurfaceNormalMatrix -> SurfaceNormalMatrix,
SurfaceNumber -> SurfaceNumber, Temperature -> Temperature,
UnconfinedIncrement -> 1, UnconfinedPath -> UnconfinedPath,
UnconfinedPosition -> UnconfinedPosition, WaveLength -> 0.532}
```

We can use the following command in order to extract the raw data of each ray(tilt, position, intensity) from our system by using the following command:

- **ReadRays**[objectset, rayparameters, selectionproperties, options] is an advanced function that takes an object set composed of Ray objects and returns a list of values for rayparameters given.

Let us use this command in order to extract the ray tilt of each ray exiting the lens. For simplicity reasons, i will propagate the system with only 2 rays:

```
    ReadRays[SimpleSystem, RayTilt, ComponentNumber -> 2]:
   {{0.962427, 0.271539, 0}, {0.962427, -0.271539, 0}}
```

This is perhaps one of the most useful commands, that lets the user to use these data and analyze the system.

2.3 Functions created by Optica Software

Even though, the optica software has a big amount of built-in functions, i had to create my own functions, in order to further analyze the system. In the following section every single functions constructed by myself will be presented. The complete coding of the following functions can be found in Appendix.4.2.

FindFocusPoint3D[OptSystem]

FindFocusPoint3D is a function that finds the paraxial focus point of the system in 3D. It treats the system as a black box and it propagates 2 paraxial rays and finds the point of intersection. It takes as an input the name of the optical system the user wants to analyze.

```
FindFocusPoint3D[SimpleSystem]:
"FocalPoint" -> {203.524, 0., 0.}
```

PlaneCreateFree[OptSystem,X]

PlaneCreateFree is a function that finds a plane in 3D space. Its center is the value X and the vertical vector of the plane is [x,y,z]=[1,0,0]. This simple function will be later used in the LinePlaneIntersection command.

```
PlaneCreateFree[SimpleSystem, 150]:
x == 150
```

LinePlaneIntersection[OptSystem,X]

LinePlaneIntesection is a function that finds all the points of intersection of the rays with the plane that the user has chosen at a distance X. Once again, for simplicity reasons, we will propagate 4 rays instead of 20.

```
LinePlaneIntersection[SimpleSystem, 150]:
{{150., -13.0639, 0.}, {150., -4.44657, 0.}, {150.,
   4.44657,0.},{150., 13.0639, 0.}}
```

CircularRayGrid[width,Options]

CircularRayGrid[width, options] creates a set of circular rays. First it begins by creating a square grid with a Ray number of NxN, where N is defined by the NumberOfRays $-$ > N command. Then it cuts the rays that have a distance greater than the input width. And thus a circular grid is created. The rays on this grid have the same intensity value.

Lets us define another simple system to visuallize this grid:

```
SimpleSystem2 = PropagateSystem[{
Move[CircularRayGrid[50, NumberOfRays -> 20], {0, 0, 0}],
Move[PlanoConvexLens[100, 50, 10], {100.0, 0, 0}],
Boundary[300]
}];
```

Once again in order to visuallize the optical system, we use the ShowSystem command. This time as a plot type we use the Full3D option, to understand better the grid.

```
XShowSystem[SimpleSystem2, PlotType -> Full3D, Axes -> True
   , Boxed -> False, AxesLabel -> {x, y}]
```

The result can be seen in Fig.2.2.

SystemSpotSizeFree[OptSystem,X]

SystemSpotSizeFree[OpticalSystem, X] is a command that finds the spot size, taking into consideration the intensity of each ray, at the distance X.

Because it does not make sense to talk about a spot size in a 2 dimensions(only for a spot diameter), we will use the SimpleSystem2 that we created with the circular grid.

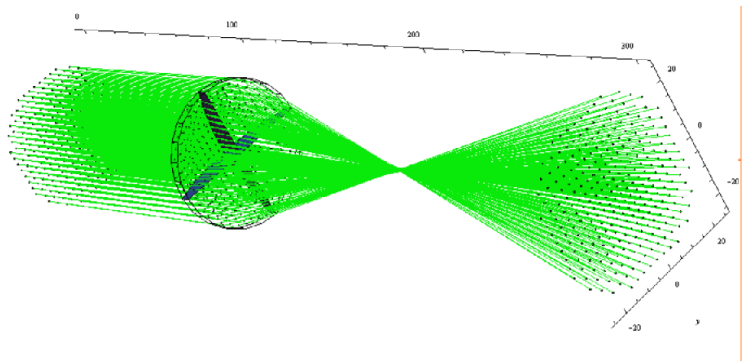


Figure 2.2: A plano convex lens system with a circular ray grid passing through it.

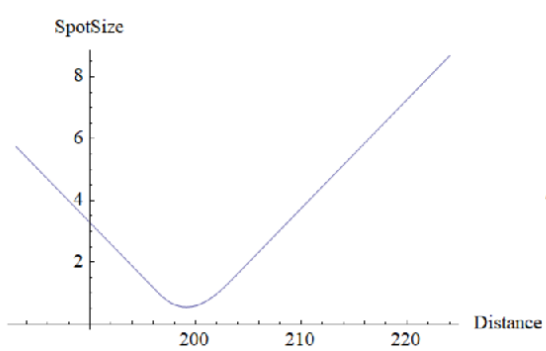
```
    SystemSpotSizeFree[SimpleSystem, 150]:
    {"SpotSize" -> 17.3697,
 "SpotPosition" -> {150., -9.18557*10^-16, 9.88581*10^-17}}
```

SystemSpotSizeFocus[OptSystem]

SystemSpotSizeFocus is a function that finds the focus point by scanning the area close to the paraxial focus and then finding the minimum value of the spot size. The output of the function gives to the user several information such as the total scanning area, the step that was used, the initial Focus point estimation and the correct focus point. It also gives a plot in order to check the results.

```
    SystemSpotSizeFocus[SimpleSystem2]:
```

```
{SpotSize of the system → 0.54861,
 SpotSize coordinates → {199.135, -4.38893, -4.38893},
```




```
, Total Area Scanning → 41.0136,

Initial FocusPoint Estimation → {203.524, 0., 0.}, Step → 0.0315418}
```

Figure 2.3: The fucntion SystemSpotSizeFocus used in the SimpleSystem2.

SystemSpotSizeFocusCustom[OptSystem,X,step]

SystemSpotSizeFocusCustom is the same function with the SystemSpotSizeFocus with the only difference being that the user picks an area to be scanned and also the step of scanning.

This function is rather useful when the user already knows the system he is testing. The output of the function remains the same with the SystemSpotSizeFocus one.

GaussianIntensity[Distance,W]

GaussianIntensity is a function that calculated the intensity of a ray depending of its distance from the optical axis. This simple function will be later used in the GaussianRayGrid in order to calculate each ray intensity depending on its position. W is the semidiamter of the beam.

GaussianRaysGrid[Width,W,options]

GaussianRayGrid[width, options] creates a set of gaussian rays. First it begins by creating a square grid with a Ray number of NxN, where N is defined by the NumberOfRays $->$ N command. Each Ray has a custom intensity value calculated by the GaussianIntensity command. Then it cuts the rays that have a distance greater than the input width. It must be used with a ThresholdIntensity $->0$ when the system is defined. W is the semi-diameter of the beam.

The ThesholdIntensity is a value for which the system does not take into consideration the rays that have a value of intensity below the Threshold one. For example if Threshold-Intensity $->5$, then the rays with a intensity value below 5 will not be propagated.

This GaussianRayGrid function is the same with the CircularRayGrid with the only difference being the intenisty of each ray. Similarly we will get the same illustration of the system seen in Fig.2.2.

LongitudinalAberration[OptSystem,Diameter]

LognitudinalAberration[OptSyst,Diameter] is a function that finds the longitudinal aberration of the system. The output is a plot that shows the longitudinal aberration in relation to the normalized pupil. The user can choose the diameter of the area tested or use the Automatic option that analyses the whole area of the system.

```
LonitudinalAberration[SimpleSystem,Automatic]:
```

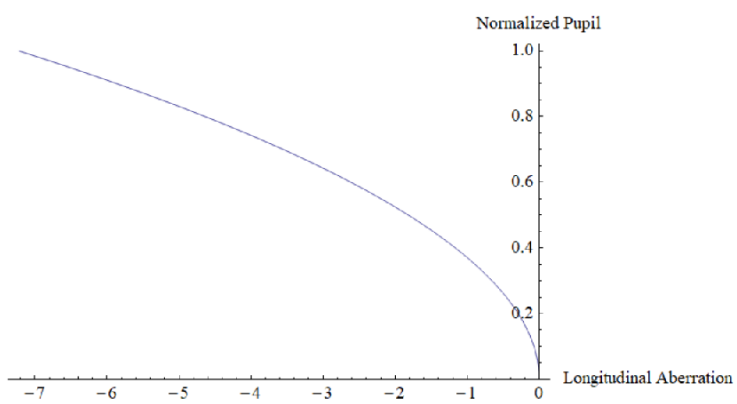


Figure 2.4: Longitudinal aberration of the simple system.

TransverseAberration[OptSystem,Diameter]

TransverseAberration[OptSyst] is a function that finds the transverse aberration of the system. The output is a plot that shows the transverse aberration in relation to the normalized

pupil. The user can choose the diameter of the area tested or use the Automatic option that analyses the whole area of the system.

```
TransverseAberration[SimpleSystem,Automaric]:
```

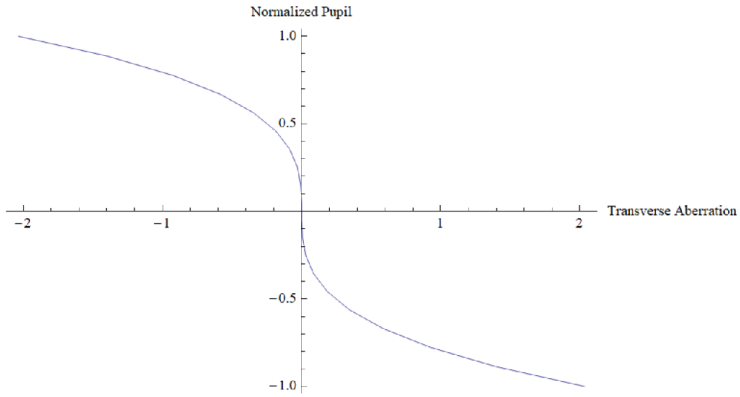


Figure 2.5: Transverse aberration of the simple system.

SpotDiagramSystem[OptSystem,Diameter]

SpotDiagramSystem is a function that constructs a plane at the paraxial focus and visualizes the spot size of the marginal and half-marginal rays. It also gives a plot that gives the user a closer look at what is happening near the paraxial point. The red line represents the paraxial focus plane. The user can choose the diameter of the area tested or use the Automatic option that analyses the whole area of the system.

```
SpotDiagramSystem[SimpleSystem,Automatic]:
```

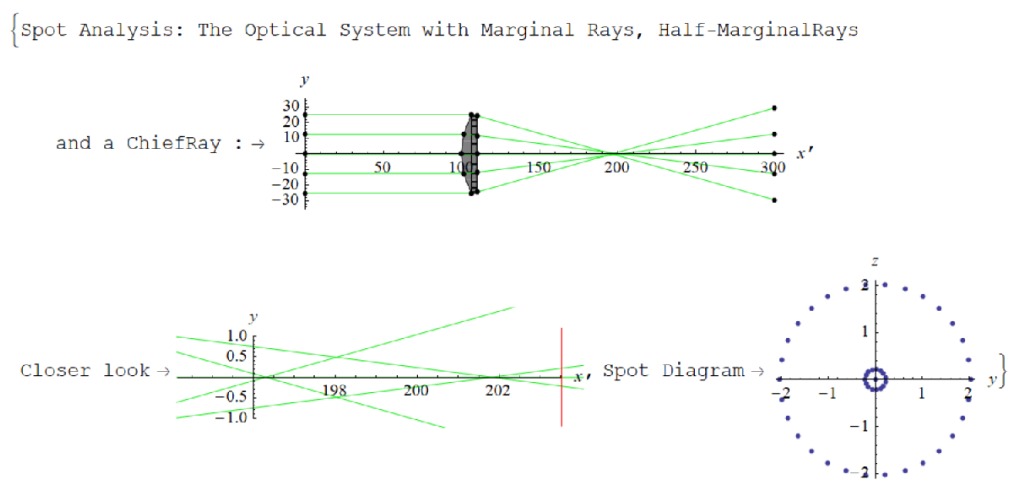


Figure 2.6: The SpotDiagramSystem function of the simple system.

SphericalAberrationPackage[OptSystem,Diameter]

SphericalAberrationPackage is a total system analysis function that shows the longitudinal and transverse aberration of the system, the spot diagram of the marginal and half marginal

rays, and a closer look of these rays compared to the paraxial ones.The user can choose the diameter of the area tested or use the Automatic option that analyses the whole area of the system. It basically sums up the LongitudinalAberration, TransverseAberration and SpotDiagramSystem functions in a single one.

LensPower[n,R1,R2,t]

LensPower[m,R1,R2,t] is a simple function that calculates the power of a lens, the effective, front and back focal length and the position of the principal planes. Furthermore, "n" is the refractive index of the lens, "R1" and "R2" the radius of curvature in front and at the back of the lens and "t" is the thickness of the lens.

```
    LensPower[1.5, 100, -100, 10]
    {{"Power", 0.00983333}, {"EFL", 101.695}, {"FFL", 105.085},
        {"bfl",98.3051}, {"First Principal", -3.38983}, {"
       Second Principal",
  3.38983}}
```

LensCombo[Lens1,Lens2,d]

LensCombio[Lens1,Lens2,d] is a simple function that calculates the total power of a two lens system, the effective, front and back focal length of this system and the position of the principal planes. "Lens1" and "Lens2" are defined by the LenPower function. Finally, "d" is the distance between the second principal plane of the first lens with the first principal plane of the second lens.

```
    LensCombo[Lens1, Lens2, 1.9]:
    {{"Power", 0.0333508}, {"EFL", 29.9843}, {"FFL", 30.7042},
       {"Bfl",27.499}, {"First Principal", -0.719958}, {"
       Second Principal",
  2.48528}}
```

Chapter 3

Construction of the fiber couplers

Introduction

In this chapter, we will present you, two fiber collimator systems for two different single-mode optical fibers. Those fibers are real optical fibers taken from the ThorLabs website. For more information about the ThorLabs company press here.

3.1 Sm-400 Single mode fiber collimator

Thorlabs' SM400 fiber consists of an un-doped, pure silica core surrounded by a depressed, fluorinedoped cladding. The specifications of this optical fiber, taken from the ThorLabs website are given in Talbe.3.1:

We can confirm the NA value of the fiber, by calculating it, using Eq.1.38 . We get that NA=0.13. We can now use Eq.1.53 to get :

$$d = 2NAf = 0.26f \tag{3.1}$$

The Sm400 is a step-index single mode optical fiber. Now we are ready to construct a fiber collimator for this fiber.

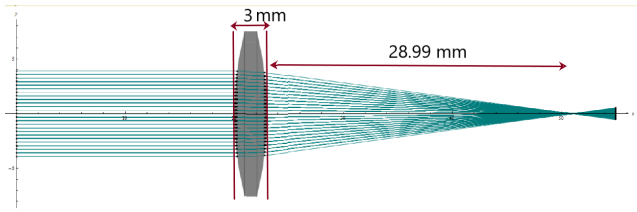
Let us begin by choosing a focal length of 30 mm. Using the previous equation, the output beam diameter will be d=7.8 mm.Now let us construct such a system. We can approach the analysis of the system by choosing the output diameter first and then finding a focal length value. Both approaches are correct and depend on the task the optical design engineer was given. Because there are no restrictions in this analysis, we chose the first approach and decided to firstly choose a lens.

Fiber Name	Sm400
Operating Wavelength	405-532 nm
M.F.D.	2.5 - 3.4 µm @ 480 nm
Cladding	125 ± 1 μm
Coating	245 ± 15 μm
NA	0.12 - 0.14
Core Index	467 nm: 1.46435
Cladding Index	467 nm: 1.45857

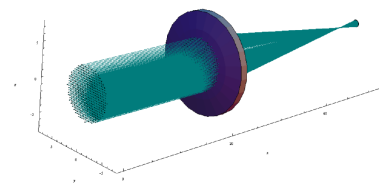
Table 3.1: Specification of the SM400 optical fiber. Taken from here.

	Comment	Radius(mm)	Thickness(mm)	Material	Semi-Diameter(mm)
1		Infinity	20	Air	7.5
2	BiConvexLens	30.4882	3	BK-7	7.5
3		-30.4882	28.99	Air	7.5

Table 3.2: Bi-convex lens specification.



(a) Rendering of the system with PlotType− >SideView.



(b) The rendering of the system with PlotType− >Full3d.

Figure 3.1: Fiber collimator using a simple Bi convex lens.

The following subsections contain different fiber collimator designs. Those designs will be tested by analysing the spherical aberration and the spot size of each system. Finally we will choose the design with the least aberration values and the smallest spot size. The task is to construct such a system that will focus the light withing the M.F.D. region. The material used for the lenses is the BK7 which is most suitable for the wavelength range of .35 microns to 2.3 microns. and has a refractive index of ≈ 1.5.

System 1: Bi Convex Lens

The simplest fiber collimator can be constructed by using a Bi Convex Lens.The specification of the lens is given in Table.3.2.

Let us define the system in Optica software:

```
System1 = PropagateSystem[{
Move[CircularRayGrid[7.8, NumberOfRays -> 25,
  WaveLength -> 0.480], {0, 0, 0}],
Move[SphericalLens[30.4882, -30.4882, 15, 3, 2,
  ComponentMedium -> BK7, DesignWaveLength -> 0.480,
  GraphicDesign -> Solid], {20, 0, 0}],
Boundary[70]
}, ThresholdIntensity -> 0];
```

The rendering of this system can be seen in Fig.3.1a. Note that we have used the CircularRayGrid function. From the Side View it looks like a line of rays but the true rays can be seen if we render the system in 3D, as seen in Fig.3.1b. In the following systems, the Side View option will be mainly used, in order to visualize the thickness of the lenses and the back focal lengths.

Using the LensPower function we can calculate the effective, front and back focal length, the power of the lens and the position of the principal planes:

```
LensPower[1.5168, 30.4882, -30.4882, 3]
{{"Power", 0.0333333}, {"EFL", 30.}, {"FFL", 31.0058}, {"
    bfl",28.9942}, {"First Principal", -1.00578}, {"Second
    Principal", 1.00578}}
```

Now let us move to testing the system. Using the SphericalAberrationPackage function, we get the following results, shown in Fig.3.2.

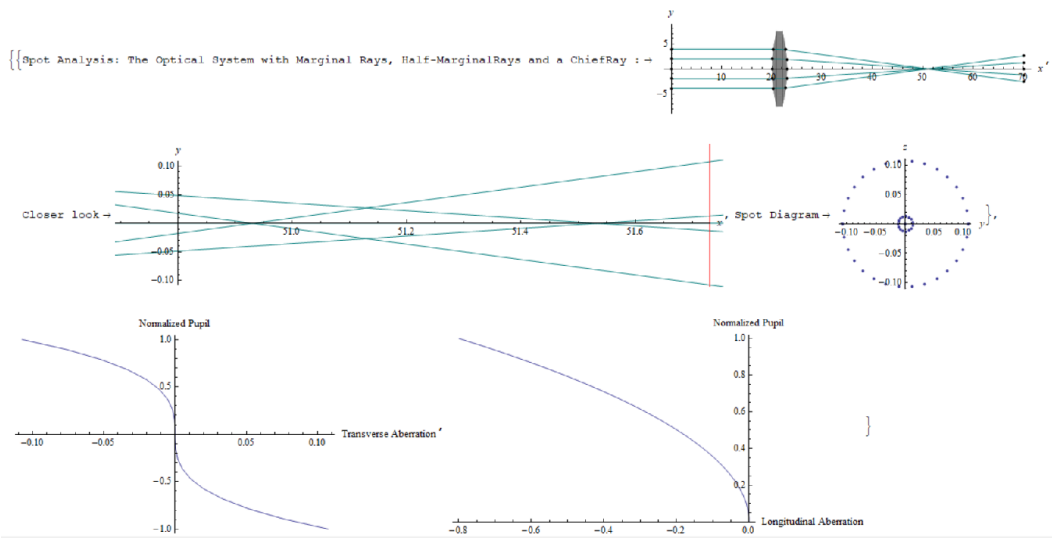


Figure 3.2: Spherical Aberration analysis of the system1.

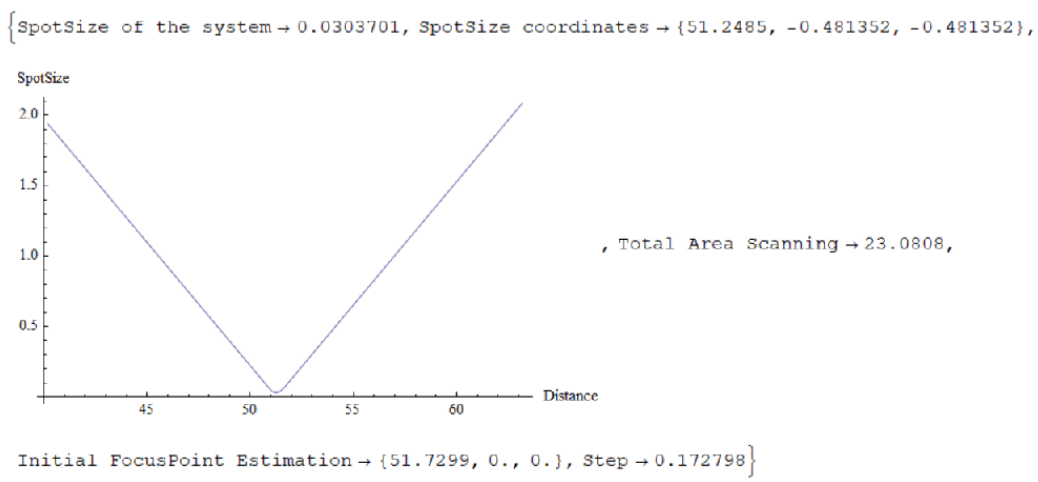


Figure 3.3: SpotSize analysis of system1.

We have tested the aberrations of this system, by focusing only on the given output beam diameter and not the whole diameter/aperture of the system. That is why we normalize the system from $-7.8/2 = -3.9$ to $7.8/2 = 3.9$. The plot of the longitudinal aberration, gives us a maximum deviation value of 0.8 mm from the paraxial focus. In other words, the rays that pass through the outermost area of the 7.8 mm, miss the paraxial point by 0.8 mm. Next we use the SystemSpotSizeFocus function to find the minimum spot size of the system. The output can be seen in Fig.3.3. All these information will be presented in a the following Table3.3 and in the Appendix4.1. The value of the spot size is S=0.03037 mm= 30.07 μm. The task is to reduce the spherical aberrations and thus the spot size by trying different systems .

This procedure will be repeated for the other systems as well.

Plano Convex Lenses

Another simple lens design is the plano convex lens. This lens has a flat side and a curved one. The two following systems will include a plano convex lens placed in two different

Fiber Name	Sm400
Design	BiConvexLens
EFL	30 mm
FFL	31.0058 mm
BFL	28.9942
Paraxial Focus	51.7299 mm
First Principal plane	-1.0057 mm
Second Principal plane	1.0057 mm
Min/Max Longitudinal Aberration	-0.8,0 mm
Min/Max Transverse Aberration	-0.10751,0.10751 mm
SpotSize	0.03037 mm
SpotSize Coordinate in X-axis	51.2485 mm

Table 3.3: Analysis data for system1

	Comment	Radius(mm)	Thickness(mm)	Material	Semi-Diameter(mm)
1		Infinity	20	Air	7.5
2	PlanoConvexLens	Infinity	3	BK-7	7.5
3		-15.504	30.0	Air	7.5

Table 3.4: Plano convex lens specification with flat side first..

ways. The first one with the flat side first and the second one with the curved side first. We will see that even though, these lenses are the same, the results of each ray tracing will be different.

System2: Plano Convex Lens with the flat side first

Let us begin with flat side first. The specification of the lens can be seen in Table.3.4:

Defining the system in Optica software:

```
System2 = PropagateSystem[{
Move[CircularRayGrid[7.8, NumberOfRays -> 20,
  WaveLength -> 0.480], {0, 0, 0}],
Move[SphericalLens[Infinity, -15.504, 15, 3, 2,
   ComponentMedium -> BK7], {20, 0, 0}],
Boundary[70]
}, ThresholdIntensity -> 0];
```

The rendering of the system can be seen in Fig.3.4.

Using the LensPower function we can calculate the effective, front and back focal length, the power of the lens and the position of the principal planes. Using the Spherical Aberration

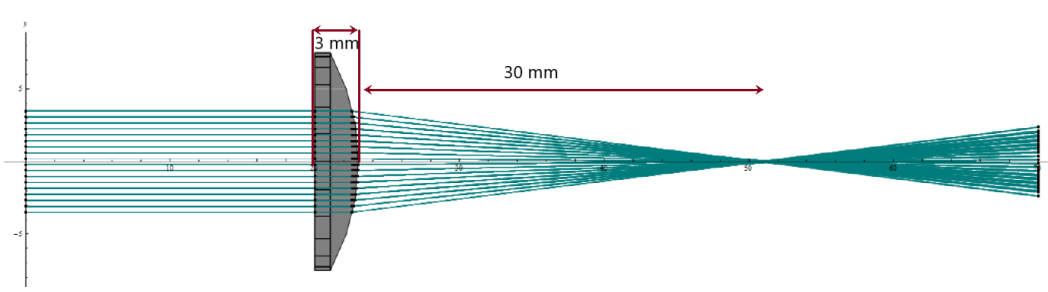


Figure 3.4: Rendering of System2.

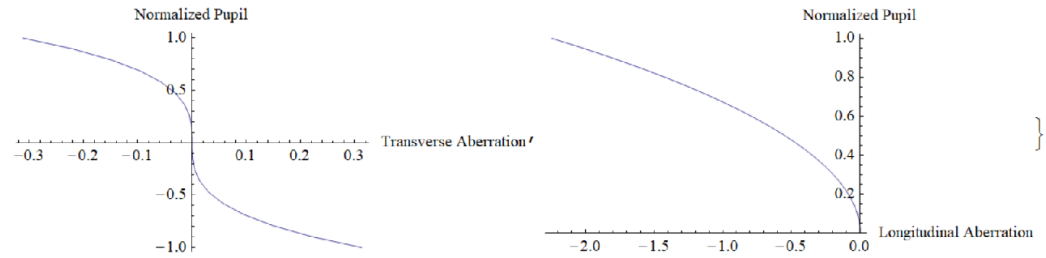


Figure 3.5: Transverse and longitudinal aberration of System2.

Fiber Name	Sm400
Design	Plano Convex Lens-Flat first
EFL	30 mm
FFL	31.9778 mm
BFL	30 mm
Paraxial Focus	52.7291 mm
First Principal plane	-1.9778 mm
Second Principal plane	0 mm
Min/Max Longitudinal Aberration	-2.24,0 mm
Min/Max Transverse Aberration	-0.3125,0.3125 mm
SpotSize	0.09201 mm
SpotSize Coordinate in X-axis	51.3291 mm

Table 3.5: Analysis data for system2

package function, we get the following result, as seen in Fig.3.5. We will show only the plots of the longitudinal and transverse aberrations. The whole output of the function can be seen in Appendix4.1.

Finally we use the SystemSpotSizeFocus in order to find the minimum spot size of the system.Once again, the whole output can be found here 4.1.

All the data of the analysis can be seen in Table.3.5.

Comparing to the Simple Bi Convex Lens design,Table.3.6:

Obviously, by choosing a plano convex lens with the flat side first, makes the system even worse and it increases the aberrations.

System3: Plano Convex Lens with curved side first

Now let us see what happens to the system, by placing the plano convex lens in the opposite way. The specification of the lens can be seen in Table.3.7

Defining the system in Optica software:

```
System3 = PropagateSystem[{
Move[CircularRayGrid[7.8, NumberOfRays -> 20,
  WaveLength -> 0.480], {0, 0, 0}],
```

Lens Type	Min/Max Longitudinal(mm)	Min/Max Transverse(mm)	SpotSize(mm)
BiConvexLens	-0.8/0	-0.10751/0.10751	0.03037
PlanoConvexLens,Flat first	-2.24/0	-0.3125/0.3125	0.09201
Absolute Percentage Deviation	+180 % /0%	+190% / +190%	+202%

Table 3.6: System 2 compared to system 1

	Comment	Radius(mm)	Thickness(mm)	Material	Semi-Diameter(mm)
1		Infinity	20	Air	7.5
2	PlanoConvexLens	15.504	3	BK-7	7.5
3		Infinity	28.0222 mm	Air	7.5

Table 3.7: Plano convex lens specification with curved side first..

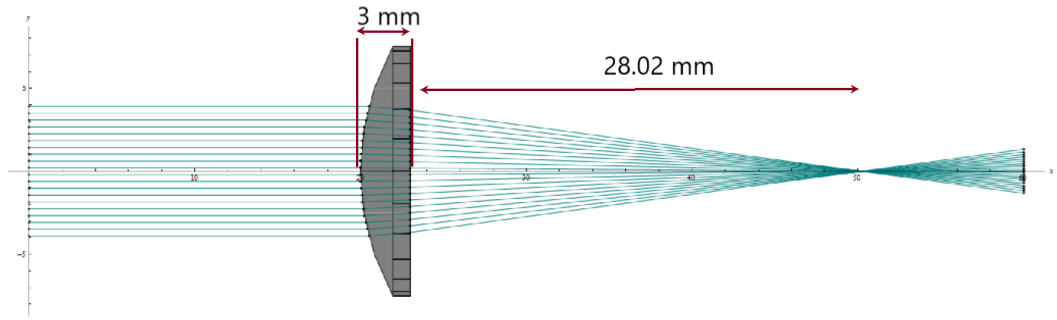


Figure 3.6: Rendering of System3.

```
    Move[SphericalLens[15.504, Infinity, 15, 3,
      ComponentMedium -> BK7], {20, 0, 0}],
    Boundary[60]
    }, ThresholdIntensity -> 0];
```

The rendering of the system can be seen in Fig.3.6

The Transverse and Longitudinal plots of the system can be seen in Fig.3.7

The whole outputs of the function used can be seen analytically in Appendix4.1. All the data of the analysis can be seen in Table.3.8.

Comparing this system to the simple Bi Convex Lens in system1, Table.3.9:

As we can see , the position of the plano convex lens is crucial when dealing with aberration. It is always better to place the curved side of the lens first. By using this design, we managed to reduce the aberration of the system by a factor of 30% in every aspect.

System4: Custom Spherical Lens with radio of curvatures equal to 1/6

We can alternate the design of the Bi Convex Lens, by changing the radius of the curvatures, by making the ratio of them equal to 1/6:

The specification of the system can be seen in Table.3.10.

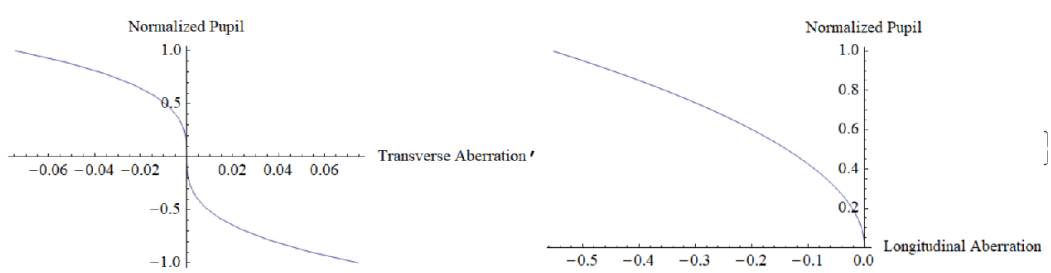


Figure 3.7: Transverse and Longitudinal aberration of System3

Fiber Name	Sm400
Design	Plano Convex Lens-Curved first
EFL	30 mm
FFL	30 mm
BFL	28.0222 mm
Paraxial Focus	50.758 mm
First Principal plane	0 mm
Second Principal plane	1.9778 mm
Min/Max Longitudinal Aberration	-0.5521,0 mm
Min/Max Transverse Aberration	-0.07466,0.07466 mm
SpotSize	0.02136 mm
SpotSize Coordinate in X-axis	50.418 mm

Table 3.8: Analysis data for system3

Lens Type	Min/Max Longitudinal(mm)	Min/Max Transverse(mm)	SpotSize(mm)
BiConvexLens	-0.8/0	-0.10751/0.10751	0.03037
PlanoConvexLens,Curved first	-0.5521/0	-0.07466/0.07466	0.02136
Absolute Percentage Deviation	-30 % /0%	-30% / -30%	-29%

Table 3.9: System 3 compared to system 1

	Comment	Radius(mm)	Thickness(mm)	Material	Semi-Diameter(mm)
1		Infinity	20	Air	7.5
2	Custom Lens	17.94	3	BK-7	7.5
3		-107.64	28.2895 mm	Air	7.5

Table 3.10: Custom lens specification.

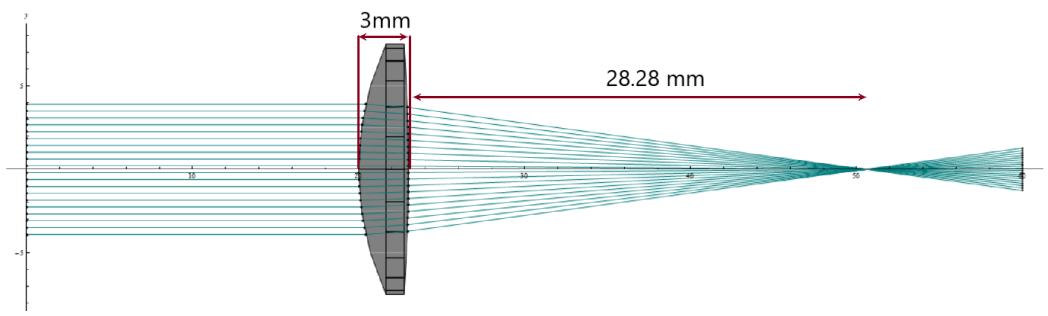


Figure 3.8: Rendering of System4

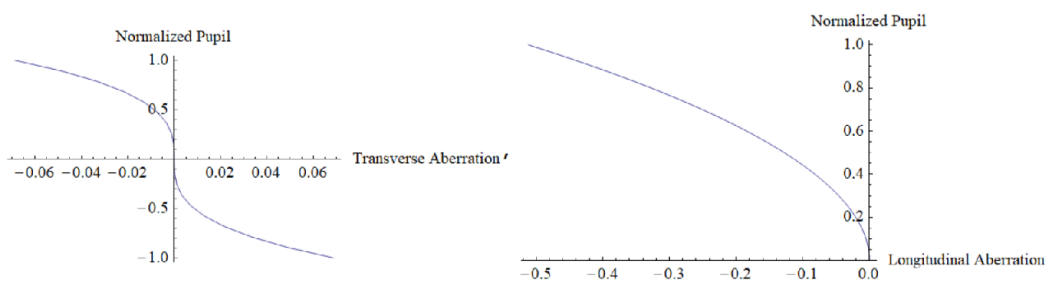


Figure 3.9: Transverse and Longitudinal aberration of System4

Now, let us define the system in Optica software:

```
System4 = PropagateSystem[{
Move[CircularRayGrid[7.8, NumberOfRays -> 20, WaveLength ->
    0.480], {0, 0, 0}],
Move[SphericalLens[17.94, -107.64, 15, 3, ComponentMedium
   -> BK7,
  DesignWaveLength -> 0.480], {20, 0, 0}],
Boundary[60]
}, ThresholdIntensity -> 0];
```

The rendering of the system can be seen in Fig.3.8.

The Transverse and Longitudinal aberrations plots of the system are shown in Fig.3.9

The whole outputs of the functions used can be seen analytically in Appendix4.1. All the data of the analysis are gathered in Table.3.11.

Now, lets us compare it with the simple Bi-Convex Lens system, Table3.12:

As we can see, this lens design, help us to reduce even more the spherical aberration. Compared the plano convex lens system, Table3.9, we see that the aberration, was further descreased by 6%.

System5: Ball Lens

Instead of a typical convex lens, a ball lens can be chosen for collimating a beam from a fiber[7]. The specification of the system can be seen in Table.3.13.

Now let us define the system in Optica Software:

```
System5 = PropagateSystem[{
Move[CircularRayGrid[7.8, NumberOfRays -> 20, WaveLength ->
    0.480], {0, 0, 0}],
Move[BallLens[40.8862], {20, 0, 0}],
```

Fiber Name	Sm400
Design	Custom Lens first
EFL	30 mm
FFL	30.2836 mm
BFL	28.2895 mm
Paraxial Focus	51.0259 mm
First Principal plane	-0.2848 mm
Second Principal plane	1.7092 mm
Min/Max Longitudinal Aberration	-0.5116,0 mm
Min/Max Transverse Aberration	-0.06883,0.06883 mm
SpotSize	0.01948 mm
SpotSize Coordinate in X-axis	50.725 mm

Table 3.11: Analysis data for system4

Lens Type	Min/Max Longitudinal(mm)	Min/Max Transverse(mm)	SpotSize(mm)
BiConvexLens	-0.8/0	-0.10751/0.10751	0.03037
Custom Lens	-0.5116/0	-0.06883/0.06883	0.01948
Absolute Percentage Deviation	-36 % /0%	-35.9% / -35.9%	-35.8%

Table 3.12: System 4 compared to system 1

	Comment	Radius(mm)	Thickness(mm)	Material	Semi-Diameter(mm)
1		Infinity	20	Air	20.44
2	BallLens	20.4431	40.8862	BK-7	20.44
3		-20.4431	9.55699 mm	Air	20.44

Table 3.13: Ball lens system specification.

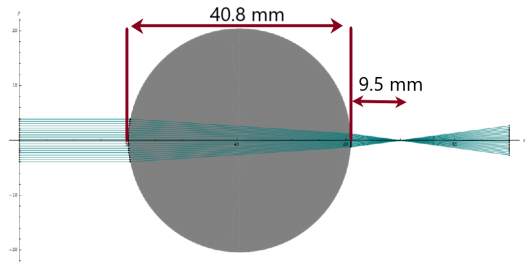


Figure 3.10: Rendering of System5

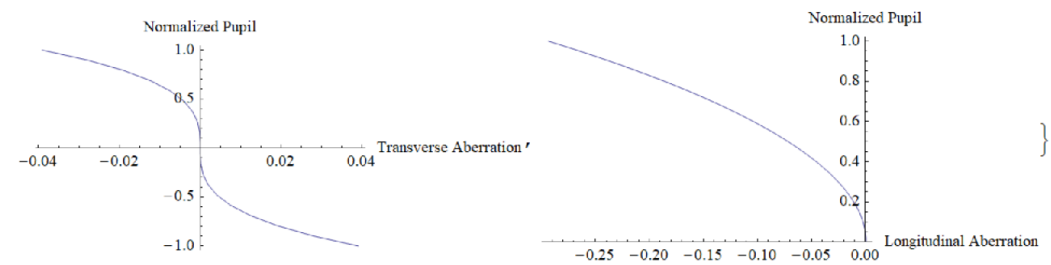


Figure 3.11: Transverse and Longitudinal aberration of system5.

```
    Boundary[90]
    }, ThresholdIntensity -> 0];
```

The rendering of the system can be seen in Fig.3.10.

The Transverse and Longitudinal aberration plots can be seen in Fig.3.11.

The whole outputs of the functions used can be seen analytically in Appendix4.1. All the data aquired from the analysis are found in Table.3.14.

Comparison of Ball Lens system (System5) with the simple Bi-Convex Lens system (System1), Table.3.15:

This specific ball lens design has further decreased the spherical aberration compared to the previous system, Table.3.11 by 33 %.

Fiber Name	Sm400
Design	Ball Lens
EFL	30 mm
FFL	50.4432 mm
BFL	9.5569 mm
Paraxial Focus	70.2643 mm
First Principal plane	-20.4431 mm
Second Principal plane	20.4431mm
Min/Max Longitudinal Aberration	–0.2936,0 mm
Min/Max Transverse Aberration	-0.03913,0.03913 mm
SpotSize	0.01104 mm
SpotSize Coordinate in X-axis	70.0843 mm

Table 3.14: Analysis data for system5

Lens Type	Min/Max Longitudinal(mm)	Min/Max Transverse(mm)	SpotSize(mm)
BiConvexLens	-0.8/0	-0.10751/0.10751	0.03037
Ball Lens	-0.2936/0	-0.03913/0.03913	0.01104
Percentage Deviation	-63.3 % /0%	-63.6% / -63.6%	-63.6%

Table 3.15: System 5 compared to system 1

	Comment	Radius(mm)	Thickness(mm)	Material	Semi-Diameter(mm)
1		Infinity	20	Air	7.5
2	PlanoConvexLens 1	28.6	3	BK-7	7.5
3		Infinity	4.977	Air	7.5
4	PlanoConvexLens 2	30.8	3	BK-7	7.5
5		Infinity	27.2897	Air	7.5

Table 3.16: Two plano convex lenses system specification.

System6: Two plano convex lenses

We saw that a plano convex lens is a solid option with reduced spherical aberration. We can split the system in two plano convex lenses, that together have the same optical power. The specification of the system can be seen in Table.3.16.

Construction of the system in Optica software:

```
System6 = PropagateSystem[{
Move[CircularRayGrid[7.8, NumberOfRays -> 20,
  WaveLength -> 0.480], {0, 0, 0}],
Move[SphericalLens[28.6, Infinity, 15, 3,
  ComponentMedium -> BK7], {20, 0, 0}],
Move[SphericalLens[30.8, Infinity, 15, 3,
  ComponentMedium -> BK7], {27.977, 0, 0}],
Boundary[70]
}, ThresholdIntensity -> 0];
```

The rendering of the system can be seen in Fig.3.12.

The Transverse and Longitudinal aberration plots are shown in Fig,3.13.

The whole outputs of the functions used can be seen analytically in Appendix.4.1. All the data collected from the analysis are given from the Table.3.17.

The comparison of this system with the simple Bi Convex Lens one (System1), Table.3.3 is shown in Table.3.18.

Finally with this design, we approach our spot size goal, which is approximately 1 $\mu m = 10^{-}3mm$. "Splitting" one lens into two has helped us to reduce the aberrations produces. Compared to the ball lens, we saw another reduction of the spherical aberration by 11

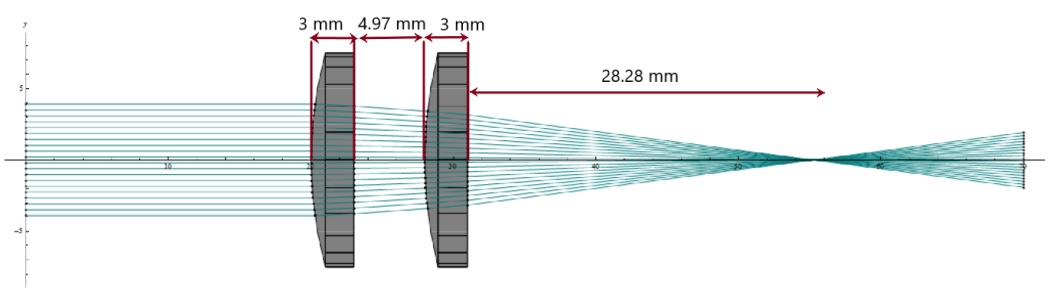


Figure 3.12: Rendering of System6

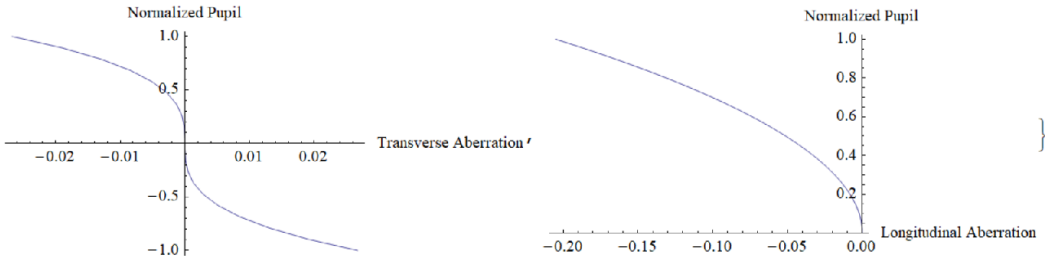


Figure 3.13: Transverse and Longitudinal aberration of System6

Fiber Name	Sm400
Design	Two Plano Convex Lenses
EFL	30 mm
FFL	27.4833 mm
BFL	27.2897 mm
Paraxial Focus	55.4523 mm
First Principal plane	2.5169 mm
Second Principal plane	2.7105 mm
Min/Max Longitudinal Aberration	-0.2052,0 mm
Min/Max Transverse Aberration	-0.02673,0.02673 mm
SpotSize	0.00757 mm
SpotSize Coordinate in X-axis	55.3223 mm

Table 3.17: Analysis data for system6

Lens Type	Min/Max Longitudinal(mm)	Min/Max Transverse(mm)	SpotSize(mm)
BiConvexLens	-0.8/0	-0.10751/0.10751	0.03037
Two Plano Convex Lenses	-0.2052/0	-0.02673/0.02673	0.00757
Absolute Percentage Deviation	-74.3 % /0%	-75.1% / -75.1%	-75%

Table 3.18: System 6 compared to system 6

	Comment	Radius(mm)	Thickness(mm)	Material	Semi-Diameter(mm)
1		Infinity	20	Air	7.5
2	Custom Lens 1	17	3.5	BK-7	7.5
3		-32	1.79	Air	7.5
4	Custom Lens 2	-17.5	2	BK-7	7.5
5		-32.79	27.499	Air	7.5

Table 3.19: Specification of the final system

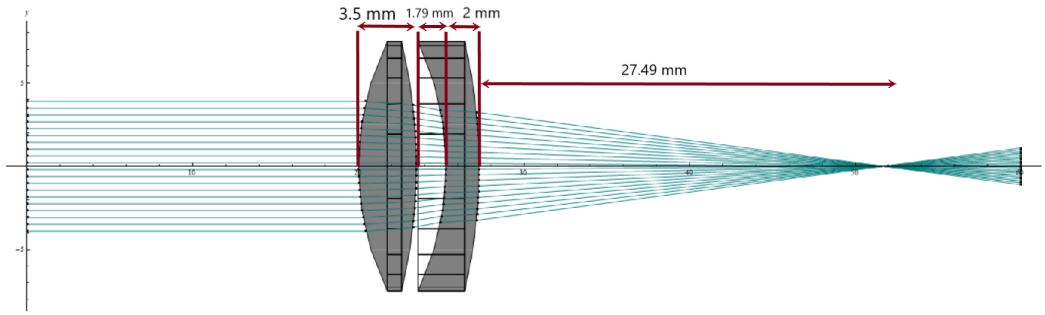


Figure 3.14: Rendering of System7

System7: Air spaced Doublet lens

This is the final design of the fiber collimator. As we will see, this system eliminates the spherical aberration, given the specific parameters of the task, which is focusing the light in a area with diameter equal to 1 μm. The following system consists of a positive power lens (Convex lens) and a negative power lens (Concave lens), which combined cancel out their individual aberrations. This technique was previously mentioned the the subsection about the spherical aberrations.

The specification of the final system, shown at Table.3.19:

Let us define the system in Optica software:

```
System7 = PropagateSystem[{
Move[CircularRayGrid[7.8, NumberOfRays -> 20,
  WaveLength -> 0.480], {0, 0, 0}],
Move[SphericalLens[17, -32, 15, 3.5, ComponentMedium -> BK7
   ], {20,0, 0}],
Move[SphericalLens[-17.5, -32.79, 15, 2,
  ComponentMedium -> BK7], {25.29, 0, 0}],
Boundary[80]
}, ThresholdIntensity -> 0];
```

The rendering of the final fiber collimator can be seen in Fig.3.14

The Transverse and Longitudinal aberration plots of the final system are shown in Fig.3.15.

The whole outputs of the functions used can be seen in Appendix.4.1. All the data collected from the analysis are given from Talbe.3.20.

Now let us compare the final fiber collimator with the first system that we introduced that was the Bi Convex Lens, Table.3.21:

We can now confirm that such a fiber collimator system is suitable for this optical fiber. The spot size of this system is $\approx 0.76\mu m$ which is below the 1 μm goal that was set in the beginning,

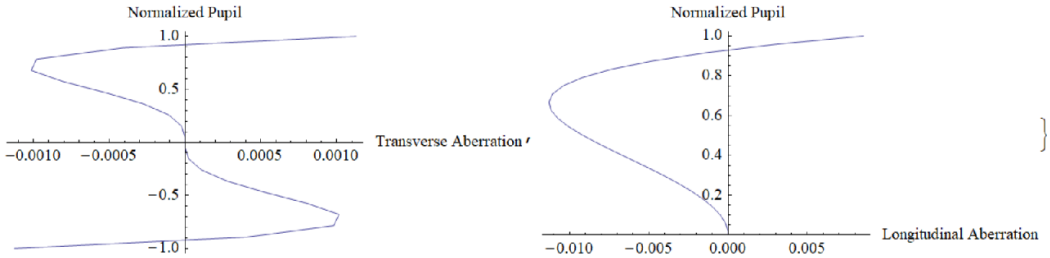


Figure 3.15: Transverse and Longitudinal Aberration of System7

Fiber Name	Sm400
Design	Air spaced Doublet
EFL	29.99 mm
FFL	30.7042 mm
BFL	27.499 mm
Paraxial Focus	51.693 mm
First Principal plane	-0.71995 mm
Second Principal plane	2.4852 mm
Min/Max Longitudinal Aberration	–0.01129, 0.00853 mm
Min/Max Transverse Aberration	-0.00112, 0.00112 mm
SpotSize	0.000758 mm
SpotSize Coordinate in X-axis	51.6873 mm

Table 3.20: Analysis data for system7

Lens Type	Min/Max Longitudinal(mm)	Min/Max Transverse(mm)	SpotSize(mm)
BiConvexLens	-0.8/0	-0.10751/0.10751	0.03037
Air Spaced Doublet	-0.01129, 0.00853	-0.00112, 0.00112	0.000758
Absolute Percentage Deviation	-78.8 % /-%	-98.9% / -98.9%	-97.5%

Table 3.21: System 7 compared to system 1

Fiber Name	SM1250G80
Operating Wavelength	1310 – 1550 nm
M.F.D.	8.2 – 9.9 μm @ 1310 nm
Cladding	80 ± 1 μm
Coating	170 ± 10μm
NA	0.11 – 0.13
Core Index	1550 nm: 1.44813
Cladding Index	1550 nm: 1.44399

Table 3.22: Specification of the SM1250G80 optical fiber. Taken from here.

	Comment	Radius(mm)	Thickness(mm)	Material	Semi-Diameter(mm)
1		Infinity	20	Air	5
2	Custom Lens 1	10	2.5	BK-7	5
3		-27	2.55	Air	5
4	Custom Lens 2	-10	-1	BK-7	5
5		-20	13.719	Air	5

Table 3.23: Specification of the SM1250G80 fiber coupler system

3.2 SM1250G80 Single mode fiber collimator

Thorlabs' specialty single mode fibers are engineered for a variety of applications including biotechnology, laser delivery, and telecommunications. These fibers offer enhanced bend-insensitivity as well as reduced splice loss while providing excellent resistance to bend induced loss similar to that of conventional fibers. The SM1250G80 offers a reduced outer diameter, which enhances resistance to mechanical failure through static fatigue, thus increasing the lifetime of the fiber. The specifications of this optical fiber, taken from the ThorLabs website are seen in Table.3.22.

We can confirm the NA value of the fiber, by calculating it, using Eq.1.38. We get that NA=0.11. We can now use Eq.1.53 to get the following relation:

$$d = 2NAf = 0.22f \tag{3.2}$$

Now let us approach the system, by choosing first an output beam diameter of 4.23 mm. Using the previous equation, we get that the effective focal length of the system must be equal to EFL=19.24 mm. Having tested multiple systems with the previous optical fiber, we can choose a similar design, just like in Fig.3.14.

Air spaced Doublet Lens

This design will contain once again an convex lens and a concave lens in order to cancel out their aberrations. The specification of this fiber coupler is shown at Table.3.23

Let us define the system in Optica software :

```
System1 = PropagateSystem[{
Move[CircularRayGrid[4.23, NumberOfRays -> 20, WaveLength
    -> 1.31], {0, 0,0}],
Move[SphericalLens[10, -27, 10, 2.5, ComponentMedium -> BK7
    ], {20,0, 0}],
Move[SphericalLens[-12.5, -22, 10, 1.5,
  ComponentMedium -> BK7], {23.08, 0, 0}],
Boundary[60]
}, ThresholdIntensity -> 0];
```

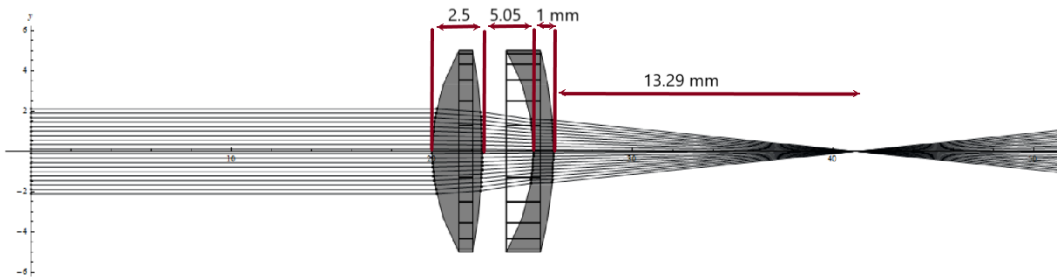


Figure 3.16: Rendering of the SM1250G80 fiber coupler system

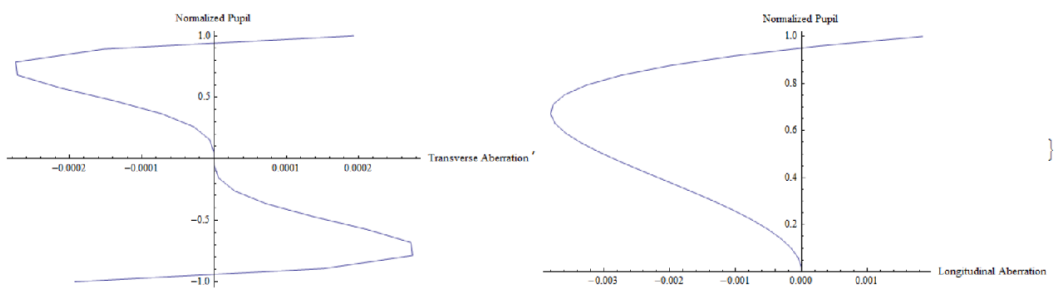


Figure 3.17: Transverse and Longitudinal aberration of the system.

The rendering of the system can be seen in Fig.3.16.

The Transverse and Longitudinal aberration plots are shown in Fig.3.17

The whole output of the functions used can be seen in Appendix4.1. All the data collected are presented in Table.3.24.

As we can see this design is a efficient one, with a spot size value of 0.13 μm which is quite small, given the area of the MFD of the fiber.

Fiber Name	SM1250G80
Design	Air spaced Doublet
EFL	19.24 mm
FFL	21.3968 mm
BFL	13.2958 mm
Paraxial Focus	41.1319 mm
First Principal plane	-2.1476 mm
Second Principal plane	5.9534 mm
Min/Max Longitudinal Aberration	–0.0038, 0.0008 mm
Min/Max Transverse Aberration	-0.00028, 0.00028 mm
SpotSize	0.00013 mm
SpotSize Coordinate in X-axis	41.1289 mm

Table 3.24: Analysis data for the SM1250G80 fiber coupler system.

Chapter 4

Annex

4.1 Full functions outputs:

This section contains the full outputs of the functions used in order to analyse the systems.

SM400: System1,Bi Convex Lens

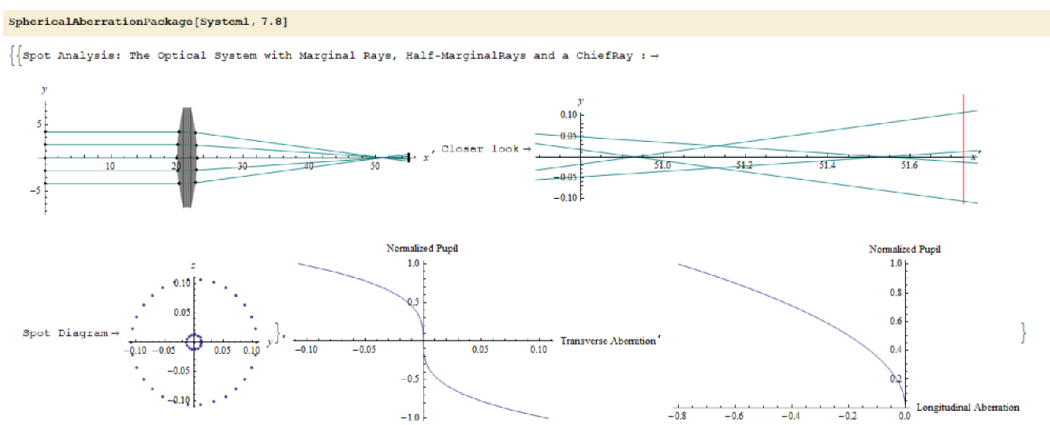


Figure 4.1: Full output of SphericalAberrationPackage of system1

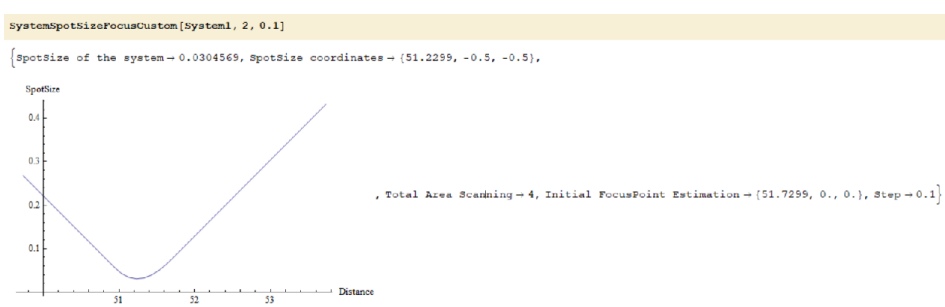


Figure 4.2: Full output of SystemSpotSizeFocus of system1

```
LensPower[1.5168, 30.4882, -30.4882, 3]
```

```
    {{"Power", 0.0333333}, {"EFL", 30.}, {"FFL", 31.0058}, {"
       bfl", 28.9942}, {"FirstPrincipal", -1.00578}, {"
       SecondPrincipal",
  1.00578}}
```

SM400: System2, Plano Convex Lens with flat side first

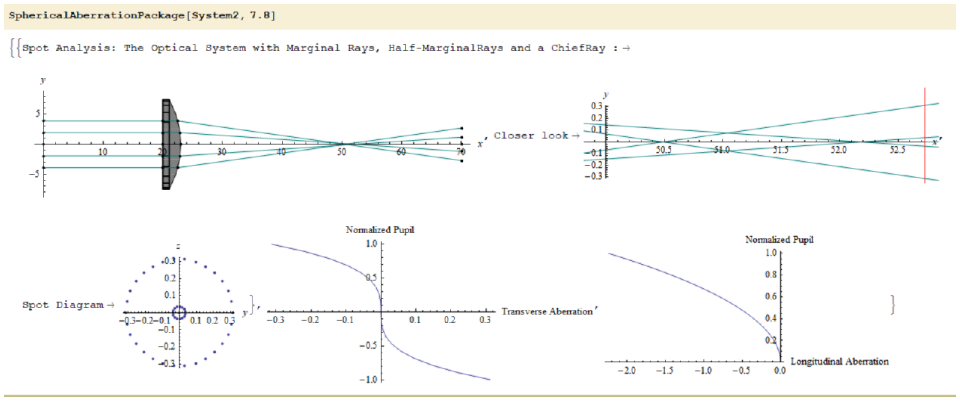


Figure 4.3: Full output of SphericalAberrationPackage of system2

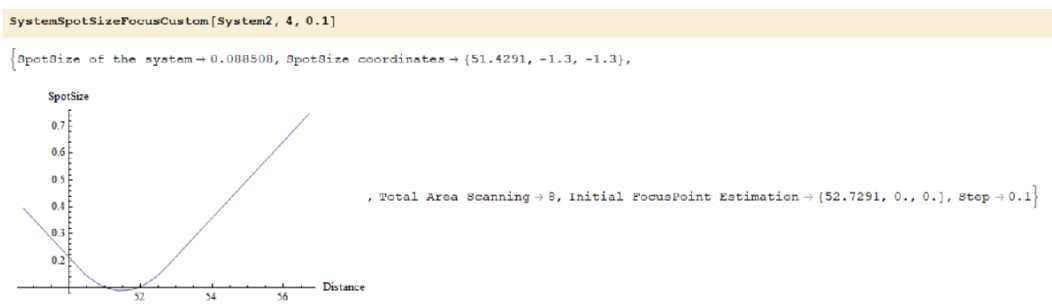


Figure 4.4: Full output of SystemSpotSizeFocus of system2

```
LensPower[1.5168, Infinity, -15.504, 3]
{{"Power", 0.0333333}, {"EFL", 30.}, {"FFL", 31.9778}, {"
    bfl", 30.}, {"First Principal", -1.97785}, {"Second
    Principal", 0.}}
```

SM400: System3, Plano Convex Lens with curved side first

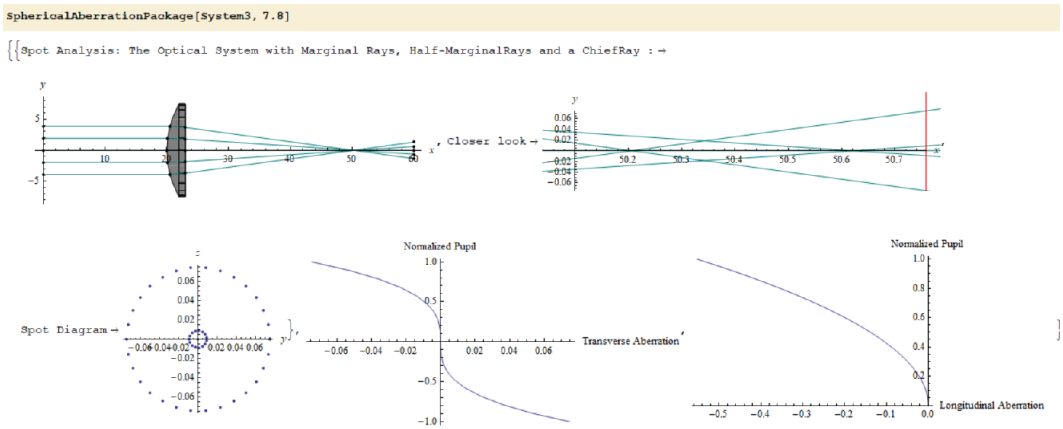


Figure 4.5: Full output of SphericalAberrationPackage of system3

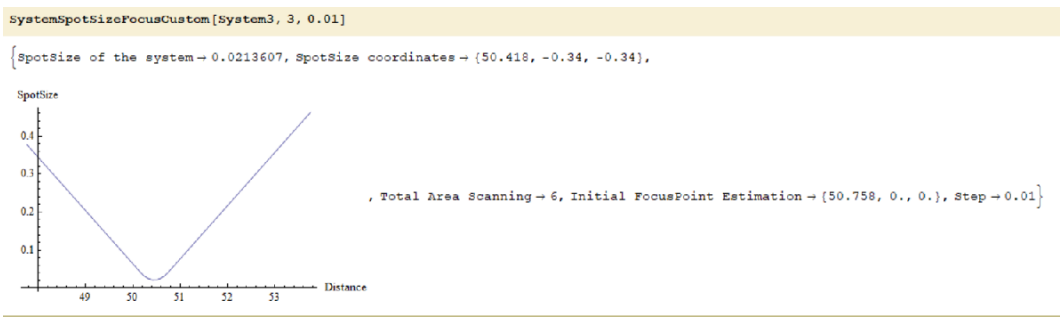


Figure 4.6: Full output of SystemSpotSizeFocus of system3

```
LensPower[1.5168, 15.504, Infinity, 3]
{{"Power", 0.0333333}, {"EFL", 30.}, {"FFL", 30.}, {"bfl"
    ,28.0222}, {"First Principal", 0.}, {"Second Principal"
    , 1.97785}}
```

SM400: System4, Crossed Lens design

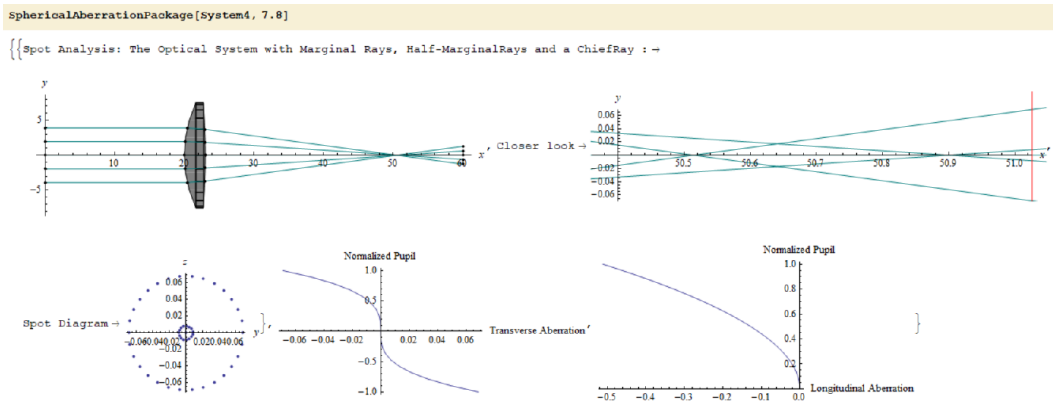


Figure 4.7: Full output of SphericalAberrationPackage of system4

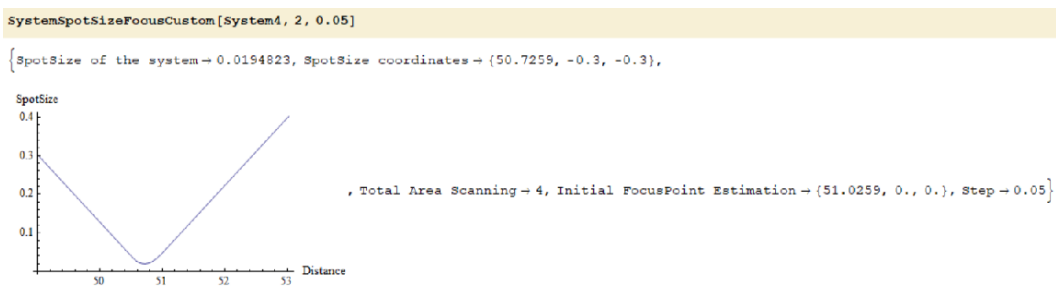


Figure 4.8: Full output of SystemSpotSizeFocus of system4

```
    LensPower[1.5168, 17.94, -107.64, 3]
    {{"Power", 0.0333348}, {"EFL", 29.9987}, {"FFL", 30.2836},
       {"bfl",28.2895}, {"First Principal", -0.284868}, {"
       Second Principal",
  1.70921}}
```

SM400: System5, Ball Lens

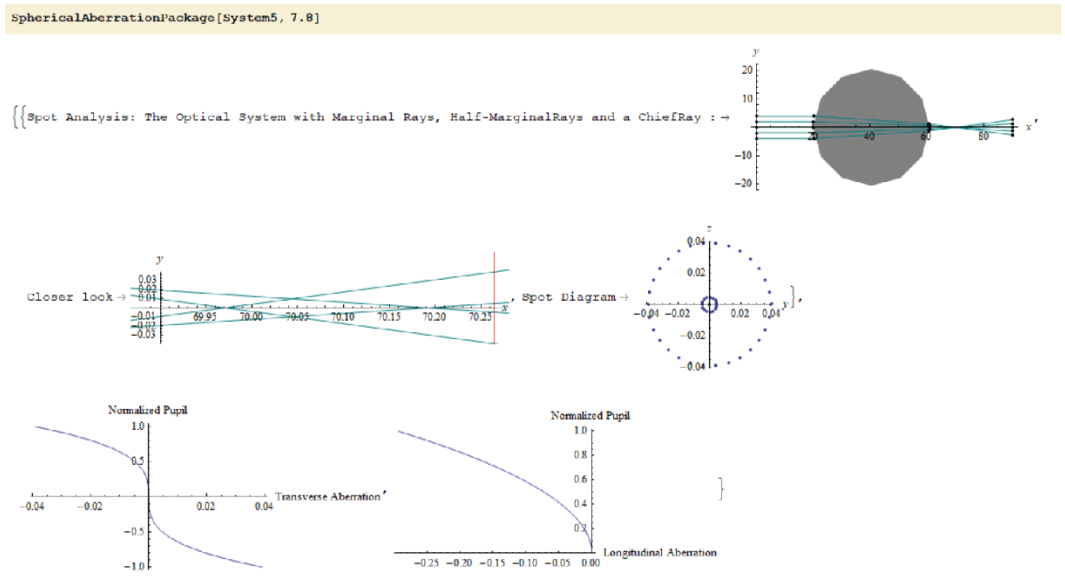


Figure 4.9: Full output of SphericalAberrationPackage of system5

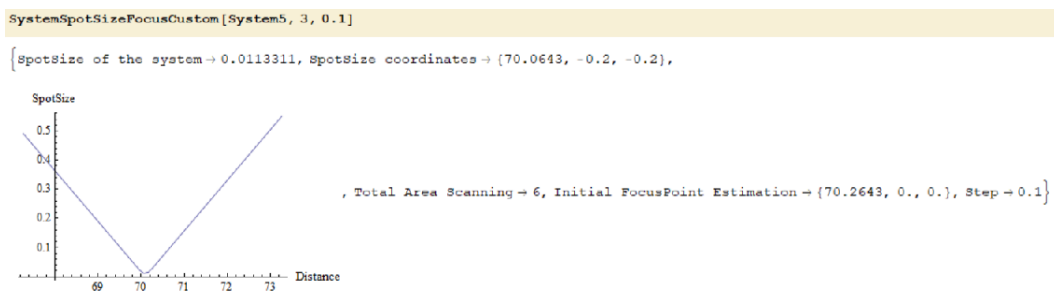


Figure 4.10: Full output of SystemSpotSizeFocus of system5

```
    LensPower[1.5168, 20.4431, -20.4431, 40.8862]
    {{"Power", 0.0333332}, {"EFL", 30.0001}, {"FFL", 50.4432},
       {"bfl",9.55699}, {"First Principal", -20.4431}, {"
       Second Principal",
  20.4431}}
```

SM400: System6, Two Plano Convex Lenses

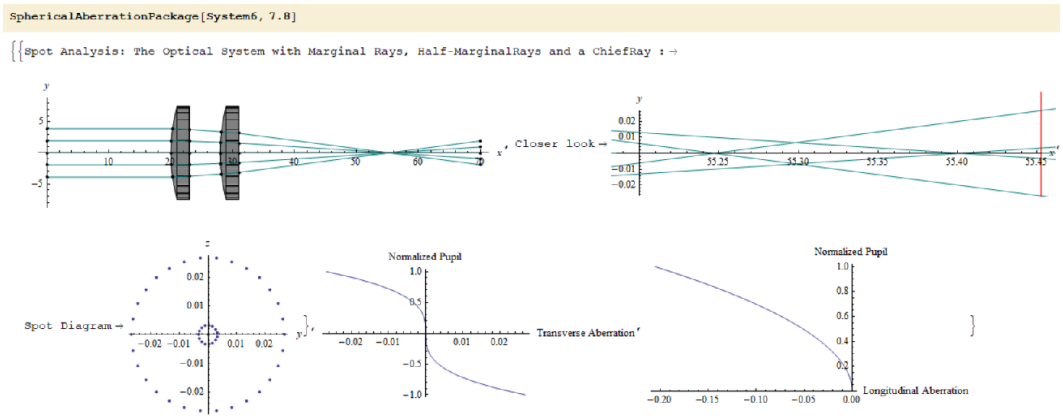


Figure 4.11: Full output of SphericalAberrationPackage of system6

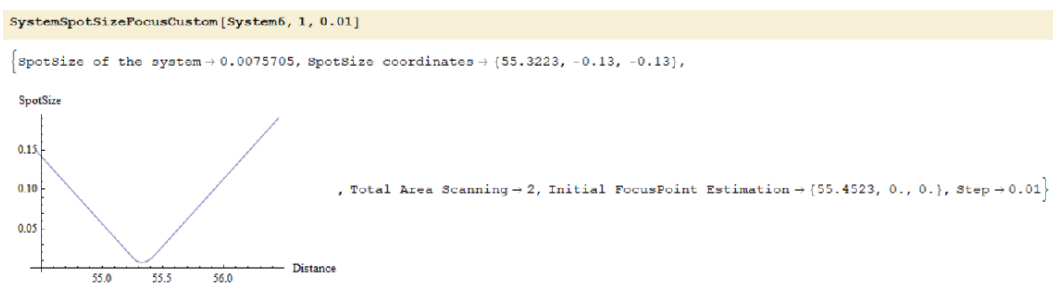


Figure 4.12: Full output of SystemSpotSizeFocus of system6

```
Lens61 = LensPower[1.5168, 28.6, Infinity, 3]
{{"Power", 0.0180699}, {"EFL", 55.3406}, {"FFL", 55.3406},
    {"bfl", 53.3627}, {"First Principal", 0.}, {"Second
    Principal", 1.97785}}
```

```
Lens62 = LensPower[1.5168, 30.8, Infinity, 3]
{{"Power", 0.0167792}, {"EFL", 59.5975}, {"FFL", 59.5975},
    {"bfl",57.6197}, {"First Principal", 0.}, {"Second
    Principal", 1.97785}}
```

```
LensCombo[Lens61, Lens62, 5]
{{"Power", 0.0333332}, {"EFL", 30.0002}, {"FFL", 27.4833},
    {"Bfl",27.2897}, {"First Principal", 2.5169}, {"Second
    Principal", 2.7105}}
```

SM400: System7, One Convex and one Concave Lens

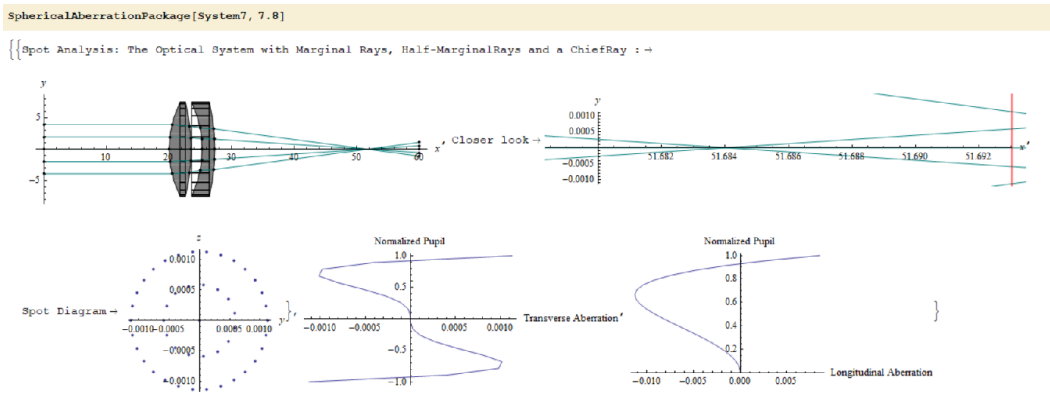


Figure 4.13: Full output of SphericalAberrationPackage of system7

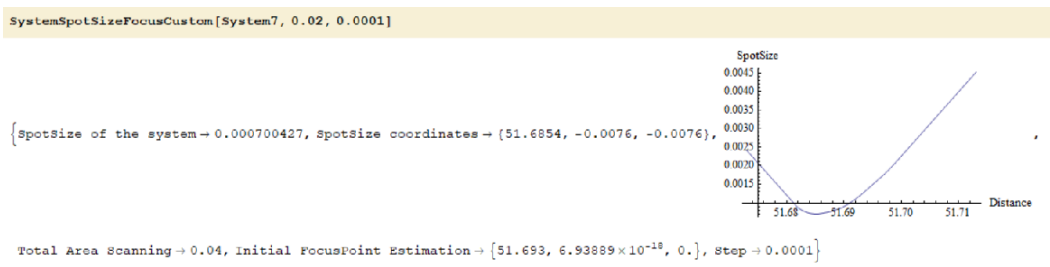


Figure 4.14: Full output of SystemSpotSizeFocus of system7

```
    Lens71 = LensPower[1.5168, 17, -32, 3.5]
    {{"Power", 0.0454171}, {"EFL", 22.0181}, {"FFL", 22.8387},
        {"bfl",20.4736}, {"First Principal", -0.820527}, {"
        Second Principal",
  1.54452}}
```

```
    Lens72 = LensPower[1.5168, -17.5, -32.79, 2]
    {{"Power", -0.0131568}, {"EFL", -76.0063}, {"FFL",
        -77.5858}, {"bfl",-78.9659}, {"First Principal",
        1.57955}, {"Second Principal",
  2.95962}}
```

```
    LensCombo[Lens71, Lens72, 1.825]
    {{"Power", 0.0333508}, {"EFL", 29.9843}, {"FFL", 30.7042},
        {"Bfl",27.499}, {"First Principal", -0.719958}, {"
        Second Principal",
  2.48528}}
```

SM1250G80: System, One Convex and one Concave Lens

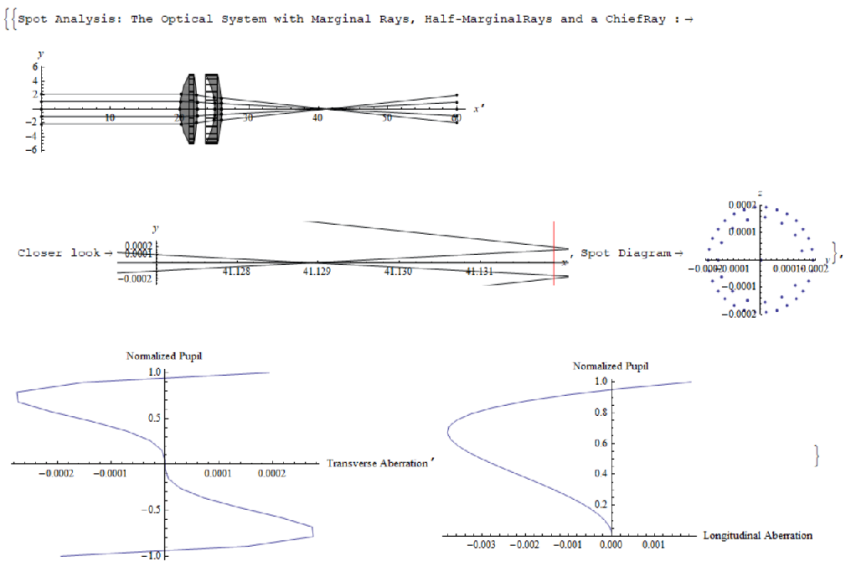


Figure 4.15: Full output of the SphericalAberrationPackage of the system

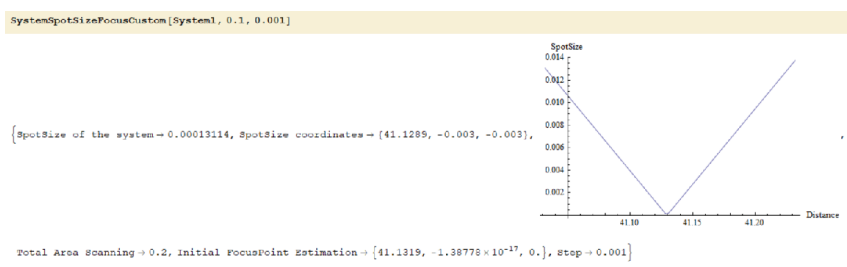


Figure 4.16: Full output of SystemSpotSizeFocus function of the system

```
    Lens1 = LensPower[1.5168, 10, -27, 2.5]
    {{"Power", 0.0691903}, {"EFL", 14.4529}, {"FFL", 14.9088},
       {"bfl", 13.2218}, {"First Principal", -0.455958}, {"
       Second Principal",
  1.23109}}
```

```
    Lens72 = LensPower[1.5168, -10, -20, 1]
    {{"Power", -0.0249596}, {"EFL", -40.0648}, {"FFL",
       -40.7473}, {"bfl",-41.4298}, {"First Principal",
       0.682538}, {"Second Principal",
  1.36508}}
```

```
    LensCombo[Lens71, Lens72, 4.47]
    {{"Power", 0.0519503}, {"EFL", 19.2492}, {"FFL", 21.3968},
       {"Bfl",13.2958}, {"First Principal", -2.14762}, {"
       Second Principal",
  5.9534}}
```

4.2 Functions coding:

This section contains all the code lines of the functions created by myself in Mathematica, using the Optica software:

FindFocusPoint3D[OptSystem]

```
    FindFocusPoint3D[OptSyst_] := Module[
  {ParaxialSystem, PropagatedSystem, x0, x1, y0, y1, z0, z1, x2
     , x3,
   y2, y3, z2, z3, dRayPos, dRayTilt, dComponentNumber,
   ParaxialRayDiameter, b, x, y, z},
  ParaxialRayDiameter =
   0.01*InputForm[OptSyst[[2, 1]][[2, 2, 2, 9, 2]]][[1]];

  %Here we create the new system that will be used for the
     propagation of paraxial Rays
  ParaxialSystem = {
    Move[LineOfRays[ParaxialRayDiameter, NumberOfRays -> 2],
       {0, 0,0}],OptSyst[[2]]
    };
  PropagatedSystem = PropagateSystem[ParaxialSystem];

  dComponentNumber = Count[PropagatedSystem[[2]], _Component];
  dRayPos =
   ReadRays[PropagatedSystem, RayStart,
    ComponentNumber -> dComponentNumber];
  dRayTilt =
   ReadRays[PropagatedSystem, RayTilt,
    ComponentNumber -> dComponentNumber];

  %Values for the first ray equation
  x0 = dRayPos[[1, 1]];
  x1 = dRayTilt[[1, 1]];
  y0 = dRayPos[[1, 2]];
  y1 = dRayTilt[[1, 2]];
  z1 = dRayPos[[1, 3]];
  z2 = dRayTilt[[1, 3]];

  %Values for the second ray equation
  x2 = dRayPos[[2, 1]];
  x3 = dRayTilt[[2, 1]];
  y2 = dRayPos[[2, 2]];
  y3 = dRayTilt[[2, 2]];
  z2 = dRayPos[[2, 3]];
  z3 = dRayTilt[[2, 3]];

  %For the following value of b the problem has a solution

  b = (x1*y0 - x0*y1 + x2*y1 - x1*y2)/(-x3*y1 + x1*y3);

  %The point of intersection is :
  x = x2 + b*x3;
```

```
  y = y2 + b*y3;
  z = z2 + b*z3;

  If[b === {}, "The system does not focus",
   "FocalPoint" -> {x, y, z}  ]
  ]
```

PlaneCreateFree[OptSystem,X]

```
PlaneCreateFree[OptSyst_, X_] :=
 Module[{Verticalvector, Pointofplane, Mainvector},

  %First we extract the distance of the plane
  Pointofplane = {X, 0, 0};

  %Next we find the vertical vector of the plane and the main vector which starts from the center of the axis and goes to the plane
  Verticalvector = {1, 0, 0};
  Mainvector = {x, y, z};

  %Finally we satisfy the following condition : Verticalvector* Pointofplance == Verticalvector* MainVector
  FullSimplify[
   Verticalvector.Pointofplane == Verticalvector.Mainvector]

  ]
```

LinePlaneIntersection[OptSystem,X]

```
LinePlaneIntersection [OptSyst_, X_] :=
 Module[{Linex, Liney, Linez, Plane, StartRay, TiltRay, t, RayCounter,
    line, Points, AllPoints, TotalPoints, i, dSRange, dopts},
  %Now lets find the point of intersection for each line equation with the plane

  %First lets create the plane using our function
  Plane = PlaneCreateFree[OptSyst, X];


  dSRange = Count[OptSyst[[2]], _Component];
  %Lets get the data from our system

  StartRay =
   ReadRays[OptSyst, RayStart, ComponentNumber  -> dSRange];
  TiltRay  = ReadRays[OptSyst, RayTilt, ComponentNumber -> dSRange];

  %Lets count the rays of the system as well
  RayCounter = Length[StartRay];
  %Now lets move on into creating the lines
```

```

 For [i = 1, i <=  RayCounter, i++,
   %Now we find the t for each ray that will be used in the general solution
   t[i] = (Plane[[2]]  - StartRay[[i, 1]])/TiltRay[[i, 1]];

   Linex[i] = StartRay[[i, 1]] + t[i]* TiltRay[[i, 1]];
   Liney[i] = StartRay[[i, 2]] + t[i]* TiltRay[[i, 2]];
   Linez[i] = StartRay[[i, 3]] + t[i]* TiltRay[[i, 3]];


   ];
%Now we find the point of intersection
 AllPoints = Table[{Linex[i], Liney[i], Linez[i]}, {i, RayCounter}]
 ]
```

SystemSpotSizeFree[OptSystem,X]

```
    SystemSpotSizeFree :: usage =
  "SystemSpotSizeFree[OpticalSystem,X] is a command that finds the \
spot size, taking into consideration the intensity of each ray, at \
the distance X.";

SystemSpotSizeFree [OptSyst_, X_] :=
 Module[{Plane, InterPoints, XAV, YAV, ZAV, CentralPoint, RayCounter,
   i, , k, r, Distance, AllDistance, AverageDistanceRadius, Else, a,
   Classic, dsRange, dComponentNumber, Normal, dRayType, F, t},



  %Now lets get all the points of intersection from the function LinePlaneIntersection function
  dComponentNumber = Count[OptSyst[[2]], _Component];
  InterPoints = LinePlaneIntersection[OptSyst, X];

  %Now lets intoduce a counter that will help us find the distance of each ray from the average value
  RayCounter = Length[InterPoints];


  %The F[i] weight function is given by the read rays intensity command
  F = ReadRays[OptSyst, Intensity,
    ComponentNumber -> dComponentNumber];
  %Now lets find the Center of the points by taking the average value of each point%
  XAV = (Sum[(InterPoints[[All, 1]][[i]])*F[[i]], {i, 1,
       RayCounter}])/Sum[F[[i]], {i, 1, RayCounter}];
```

```
  YAV = (Sum[(InterPoints[[All, 2]][[i]])*F[[i]], {i, 1,
       RayCounter}])/Sum[F[[i]], {i, 1, RayCounter}];
  ZAV = (Sum[(InterPoints[[All, 3]][[i]])*F[[i]], {i, 1,
       RayCounter}])/Sum[F[[i]], {i, 1, RayCounter}];

  CentralPoint = {XAV, YAV, ZAV};



  %Now lets calculate each distance

  r = Table[F[[t]]*InterPoints[[t]]*0.01 - CentralPoint, {t, 1,
      RayCounter}];
  AllDistance =
   Table[Sqrt[(r[[i, 2]])^2 + (r[[i, 3]])^2], {i, 1, RayCounter
      }];

  AverageDistanceRadius = (Sum[
      AllDistance[[k]], {k, 1, RayCounter}])/(RayCounter);
  {"SpotSize" -> 2*AverageDistanceRadius,
   "SpotPosition" -> CentralPoint}
  ]
```

SystemSpotSizeFocus[OptSystem]

```

SystemSpotSizeFocus [OptSyst_] :=
 Module[{Plane, InterPoints, XAV, YAV, ZAV, CentralPoint,
     RayCounter,
   i, , k, r, Distance, W, AllDistance, AverageDistance, Else,
      a,
   Classic, FocusPoint, ThetaMax, Spot, SpotCenter, Zmax, Z,
   DifferentSpotSizes, AllDifferentSpotSizes, SpotMax,
      PlaneCenter,
   dComponentNumber, MinPos, Plot1, AllZDistances, Step},

  dComponentNumber = Count[OptSyst[[2]], _Component];

  %For a start lets find where is the focus point of the system
  FocusPoint = FindFocusPoint3D[OptSyst][[2]];

  %Then lets create the plane of that point
  PlaneCenter = PlaneCreateFree[OptSyst, FocusPoint[[1]]];

  %Now the find the spot size of the focus point
  SpotCenter = SystemSpotSizeFree[OptSyst, PlaneCenter[[2]]];

  %Lets find the max angle of a ray in our system
  ThetaMax =
   Max[Abs[ArcTan[
       Sqrt[(ReadRays[OptSyst, RayTilt,
              ComponentNumber -> dComponentNumber - 1][[All,
             2]])^2 + (ReadRays[OptSyst, RayTilt,
```

```
              ComponentNumber -> dComponentNumber - 1][[All,
            3]])^2]/(ReadRays[OptSyst, RayTilt,
          ComponentNumber -> dComponentNumber - 1][[All, 1]])
             ]/
     Degree]];

%Now lets move on to the condition.We pick to scan the system
   from a spot size equal to 100 times the spot size of the
   focus point of the system

SpotMax = 100*SystemSpotSizeFree[OptSyst, FocusPoint[[1]]][[1,
    2]];
%The Zmax is the maximum distance the plane can move from the
   central plane
Zmax = 40*SpotMax/(Abs[Tan[ThetaMax  ]]);
Step = 0.1*Abs[0.3*Tan[ThetaMax  ]*SpotCenter[[1, 2]]];
%We store all the values we found. We scan from The Focus
   point -Zmax to the FocusPoint + Zmax
AllDifferentSpotSizes =
  Table[SystemSpotSizeFree[OptSyst, Z][[1, 2]], {Z,
    FocusPoint[[1]] - Zmax, Zmax + FocusPoint[[1]], Step}];

%Lets find the coordinates of the minimum spot size
MinPos =
  Position[AllDifferentSpotSizes, Min[AllDifferentSpotSizes
     ]][[1,
   1]];

AllZDistances =
  Table[Z, {Z, FocusPoint[[1]] - Zmax, Zmax + FocusPoint[[1]],
    Step}];

Plot1 = ListLinePlot[
   Join[Transpose[{AllZDistances}],
    Transpose[{AllDifferentSpotSizes}], 2],
   AxesLabel -> {"Distance", "SpotSize"}];


{"SpotSize of the system" -> Min[AllDifferentSpotSizes],
 "SpotSize coordinates" -> FocusPoint - Zmax + Step (MinPos -
    1),
 Plot1, "Total Area Scanning" -> 2*Zmax,
 "Initial FocusPoint Estimation" -> FocusPoint, "Step" -> Step
    }

]
```

LongidutinalAberration[OptSystem]

```
   LongitudinalAberration[OptSyst_] :=
Module[{RayHeight, TiltRay, dComponentNumber, Focus,
   FocusPlane,
```

```
  Angle, InterPoints, D, i, RayCounter, LA, Map, NormRayHeight,
  NewSystemRayDiameter, NewSystem, PropagatedNewSystem},

 %Making the new system, by taking its components
 NewSystemRayDiameter =
  InputForm[OptSyst[[2, 1]][[2, 2, 2, 9, 2]]][[1]];
 NewSystem = {
   Move[LineOfRays[NewSystemRayDiameter, NumberOfRays -> 50], {0, 0,
     0}],
   OptSyst[[2]]
   };
 PropagatedNewSystem = PropagateSystem[NewSystem];




 %Data extraction using the ReadRays
 dComponentNumber = Count[PropagatedNewSystem[[2]], _Component];

 RayHeight =
  ReadRays[PropagatedNewSystem, RayStart,
    ComponentNumber -> dComponentNumber][[All, 2]];
 NormRayHeight = RayHeight/Max[RayHeight];

 %First lets Find the focus point and create a plane on this point


 Focus = FindFocusPoint3D[PropagatedNewSystem][[2, 1]];
 FocusPlane = PlaneCreateFree[PropagatedNewSystem, Focus];
 %Now lets find the angle of each ray.
Angle = ArcTan[
    ReadRays[PropagatedNewSystem, RayTilt,
       ComponentNumber -> dComponentNumber][[All,
      2]]/(ReadRays[PropagatedNewSystem, RayTilt,
        ComponentNumber -> dComponentNumber][[All, 1]])]/Degree;


%Now lets calculate each Ray distance from the focus point in order to find the D distance
InterPoints =
  LinePlaneIntersection[PropagatedNewSystem, Focus][[All, 2]];

%Finding the LA
LA = -InterPoints/Tan[Angle Degree];


%Now we simply have to make a RayHeight-LA diagram
Map = ListLinePlot[
  Join[Transpose[{LA}], Transpose[{NormRayHeight}], 2],
```

```
    PlotRange -> {0, Automatic},
    AxesLabel -> {"Longitudinal Aberration", "Normalized Pupil"}]
  ]
```

TransverseAberration[OptSystem]

```
    TransverseAberration[OptSyst_] :=
  Module[{RayHeight, TiltRay, dComponentNumber, Focus, FocusPlane,
    Angle, D, i, RayCounter, LA, Map, FocusPoint, InterPoints,
    NormRayHeight, NewSystem, NewSystemRayDiameter,
    PropagatedNewSystem},

   %Making the new system, by taking its components
   NewSystemRayDiameter =
    InputForm[OptSyst[[2, 1]][[2, 2, 2, 9, 2]]][[1]];
   NewSystem = {
     Move[LineOfRays[NewSystemRayDiameter, NumberOfRays -> 20], {0, 0,
       0}],
     OptSyst[[2]]
     };
   PropagatedNewSystem = PropagateSystem[NewSystem];

   %Data extraction using the ReadRays and my functions
   dComponentNumber = Count[PropagatedNewSystem[[2]], _Component];
   FocusPoint = FindFocusPoint3D[PropagatedNewSystem][[2, 1]];
   InterPoints =
    LinePlaneIntersection[PropagatedNewSystem, FocusPoint][[All, 2]];
   RayHeight =
    ReadRays[PropagatedNewSystem, RayStart,
      ComponentNumber -> dComponentNumber][[All, 2]];
   NormRayHeight = RayHeight/Max[RayHeight];

   %Transverse Aberration map construction
   Map = ListLinePlot[
     Join[Transpose[{InterPoints}], Transpose[{NormRayHeight}], 2],
     AxesLabel -> {"Transverse Aberration", "Normalized Pupil"}]
   ]
```

SpotDiagramSystem[OptSystem]

```
    SpotDiagramSystem[OptSyst_] :=
  Module[
   {NewSystemRayDiameter, NewSystemSingleRays, PropagatedNewSys,
    PropagatedSystemSingleRays, WholePlot, ParaxialFocusPoint,
    MagnifiedPlot, ParaxialPlane, NewSystemCircleRays, SpotPlot},
```

```
%First we find the paraxial Focus
ParaxialFocusPoint = FindFocusPoint3D[OptSyst][[2]];
%Making the new system by using single Rays
NewSystemRayDiameter =
 InputForm[OptSyst[[2, 1]][[2, 2, 2, 9, 2]]][[1]];
NewSystemSingleRays = PropagateSystem[{
   Move[
    LineOfRays[0, NumberOfRays -> 1], {0, N[NewSystemRayDiameter/2],
      0}],
   Move[
    LineOfRays[0, NumberOfRays -> 1], {0, N[NewSystemRayDiameter/4],
      0}],
   Move[LineOfRays[0, NumberOfRays -> 1], {0, 0, 0}],
   Move[
    LineOfRays[0, NumberOfRays -> 1], {0,
     N[-NewSystemRayDiameter/4], 0}],
   Move[
    LineOfRays[0, NumberOfRays -> 1], {0,
     N[-NewSystemRayDiameter/2], 0}],
   OptSyst[[2]]
   }];
%First we construct the Main plot with the single Rays
WholePlot =
 XShowSystem[NewSystemSingleRays, PlotType -> TopView, Axes -> True,
   Boxed -> False, AxesLabel -> {x, y, z}];
%Then we construct another magnified plot of the system
MagnifiedPlot = Show[
  XShowSystem[NewSystemSingleRays, PlotType -> TopView,
   Axes -> True, Boxed -> False, AxesLabel -> {x, y, z},
   PlotRange -> {{ParaxialFocusPoint[[1]] +
       1.3*FullForm[
          LongitudinalAberration[OptSyst][[2, 5, 2, 1, 1]]][[1]],
      ParaxialFocusPoint[[
       1]]}, {FullForm[
        TransverseAberration[OptSyst][[2, 5, 2, 2, 2]]][[1]],
      FullForm[TransverseAberration[OptSyst][[2, 5, 2, 2, 1]]][[
       1]]}}],
  Graphics[{Red
    , Line[{{ParaxialFocusPoint[[1]],
       1.2*FullForm[
          TransverseAberration[OptSyst][[2, 5, 2, 2, 2]]][[
         1]]}, {ParaxialFocusPoint[[1]],
       1.2*FullForm[
          TransverseAberration[OptSyst][[2, 5, 2, 2, 1]]][[
         1]]}}]}]];
%Next we make another system with circle of rays in order to make the spot diagram
NewSystemCircleRays = PropagateSystem[{
   Move[
```

```
      CircleOfRays[N[NewSystemRayDiameter], NumberOfRays ->
         30], {0,
       0, 0}],
     Move[
      CircleOfRays[N[NewSystemRayDiameter/2], NumberOfRays ->
         15], {0,
        0, 0}],
     Move[LineOfRays[0, NumberOfRays -> 1], {0, 0, 0}],
     OptSyst[[2]]
     }];

 SpotPlot =
   ListPlot[
    LinePlaneIntersection[NewSystemCircleRays,
      ParaxialFocusPoint[[1]]] /. {x_, y_, z_} -> {y, z},
    AspectRatio -> Automatic, AxesLabel -> {y, z}];

 {"Spot Analysis: The Optical System with Marginal Rays, \
Half-MarginalRays and a ChiefRay :" -> WholePlot,
  "Closer look" -> MagnifiedPlot, "Spot Diagram" -> SpotPlot}

 ]
```

SphericalAberrationPackage[OptSystem]

```
    SphericalAberrationPackage[OptSyst_] :=
 Module[{Tran, Long, Spot},
  Tran = TransverseAberration[OptSyst];
  Long = LongitudinalAberration[OptSyst];
  Spot = SpotDiagramSystem[OptSyst];

  {Spot, Tran, Long}
  ]
```

CircularRayGrid[width,options]

```
    CircularRayGrid[width_, options___] :=
 Module[{h, Rays, RayPropagation, k, SquareRaysGrid,
    CircularRaysGrid,
    RayDistancey, RayDistancez, i, j, o, p, TotalRayDistance,
   OverDistanceRays, RotatedMatrixZ, RayIntensity,
      FixedRayDistance,
   GaussianAmplitude, f, FixedIntensityGrid,
      IntensitySquareRays, c,
   v},
  Rays = N[(NumberOfRays /. Flatten[{options}] /.
      Options[CircularRayGrid])];
  %First we find each ray distance from the optical Axes
  RayDistancey =
   Table[Table[(-0.5*width + i*(width/(Rays - 1))), {i, 0, Rays
       - 1,
      1}],
```

```
    {j, 0, Rays - 1, 1}];
  RayDistancez =
   Table[Table[(-0.5*width + o*(width/(Rays - 1))), {o, 0, Rays - 1,
      1}],
    {p, 0, Rays - 1, 1}];
  RotatedMatrixZ = Transpose[Reverse[RayDistancez]];

  TotalRayDistance = Sqrt[RayDistancey^2 + RotatedMatrixZ^2];

  %Now we pick the rays that have a distance larger than the circle radius and we remove them from the final table in order to get a circular grid
  OverDistanceRays =
   Select[Position[TotalRayDistance, _?(# > (width/2) &)],
    Length[#] == (2) &];

  %We make the square grid
  SquareRaysGrid =
   Table[Table[
     Move3D[Move[Ray[Intensity -> 100], {0, N[h], 0}], {0, 0,
       N[k]}, {1, 0, 0}, 0], {h, -0.5 width,
      0.5 width, (width/(Rays - 1))}], {k, -0.5 width,
     0.5 width, (width/(Rays - 1))}
    ];
  %Then we need to cut the square grid into a circular one.  We simply remove the rays with a distance value greater than the width of the input

  CircularRaysGrid =
   DeleteCases[Delete[SquareRaysGrid, OverDistanceRays], {}]


  ]
Options[CircularRayGrid] = {NumberOfRays -> 6, Intensity -> 100};
```

GaussianIntensity[Distance]

```
    GaussianIntensity[distance_, w_] :=
 Module[{d, IntensityGauss},
  IntensityGauss = N[100*E^(-distance^2/w)]
  ]
```

GaussianRayGrid[width,w,options]

```
    GaussianRayGrid[width_, w_, options___] :=
 Module[{h, Rays, RayPropagation, k, SquareRaysGrid, CircularRaysGrid,
    RayDistancey, RayDistancez, i, j, o, p, TotalRayDistance,
   OverDistanceRays, RotatedMatrixZ, RayIntensity, FixedRayDistance,
```

```
    GaussianAmplitude, f, FixedIntensityGrid, IntensitySquareRays, c,
    v},
   Rays = N[(NumberOfRays /. Flatten[{options}] /. Options[GaussianRayGrid])];
   %First we find each ray distance from the optical Axes
   RayDistancey =
    Table[Table[(-0.5*width + i*(width/(Rays - 1))), {i, 0, Rays - 1, 1}],
     {j, 0, Rays - 1, 1}];
   RayDistancez =
    Table[Table[(-0.5*width + o*(width/(Rays - 1))), {o, 0, Rays - 1, 1}],
     {p, 0, Rays - 1, 1}];
   RotatedMatrixZ = Transpose[Reverse[RayDistancez]];

   TotalRayDistance = Sqrt[RayDistancey^2 + RotatedMatrixZ^2];

   %Now we pick the rays that have a distance larger than the circle radius and we remove them from the final table in order to get a circular grid*)
   OverDistanceRays =
    Select[Position[TotalRayDistance, _?(# > (width/2) &)],
     Length[#] == (2) &];

   %We make the square grid
   SquareRaysGrid =
    Table[Table[
      Move3D[Move[
        Ray[Intensity -> GaussianIntensity[Sqrt[h^2 + k^2], w]], {0,
         N[h], 0}], {0, 0, N[k]}, {1, 0, 0}, 0], {h, -0.5 width,
      0.5 width, (width/(Rays - 1))}], {k, -0.5 width,
      0.5 width, (width/(Rays - 1))}
     ];
   %Then we need to cut the square grid into a circular one. We simply remove the rays with a distance value greater than the width of the input

   CircularRaysGrid =
    DeleteCases[Delete[SquareRaysGrid, OverDistanceRays], {}]


   ]
Options[GaussianRayGrid] = {NumberOfRays -> 6, Intensity -> 100};
```

Bibliography

[1] Arun K. Agarwal. Review of optical fiber couplers. *Fiber and Integrated Optics*, 6(1):27–53, 1987. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01468038708219344. doi:10.1080/01468038708219344.

[2] Donald Barnhart. *Optica,Designing Optics with mathematica*. More info found at Optica Software website. URL: https://www.opticasoft.com/.

[3] B. N. Begunov. *Optical Instrumentation: Theory and design*. Mir Publishers, 1988.

[4] Frank Bucholtz, Joseph M Singley, and Naval Research Laboratory. A brief introduction to core modes in optical fiber. *United States Navy*, 2021.

[5] Harvey White Francis Jenkins. *Fundamentals of Optics*. McGraw-Hill Education, 2001. Pages=159-160.

[6] Photonics Media. Fiber optics: Understanding the basics, Jul 2015. URL: https://www.photonics.com/Articles/Fiber_Optics_Understanding_the_Basics/a25151.

[7] A Mitsui, H Suzuki, T Torikai, H Takizuka, T Toma, and Y Koike. Optimization of ball lens assembly design for gi-pof optical micro-collimator. In *Proc. 23th International Conference on Plastic Optical Fibers (POF2014), Yokohama*, 2014.

[8] Duncan T Moore. Gradient-index optics: a review. *Applied Optics*, 19(7):1035–1038, 1980.

[9] Eckhardt Optics, 2017. URL: https://www.eckop.com/resources/optical-testing/measuring-aberrations/.

[10] Dr. Rüdiger Paschotta. Fiber collimators, May 2023. URL: https://www.rp-photonics.com/fiber_collimators.html.

[11] Plots created by the RP fiber calculator software *FREE VERSION*. URL: https://www.rp-photonics.com/rp_fiber_calculator.html.

[12] Warren J Smith. *Modern optical engineering: the design of optical systems*. McGraw-Hill Education, 2008.

[13] Dimitrios Zevgolis. *Applied Optics: Fiber optics and Laser*. tziola publisher, 2016.