Theoretical, experimental and numeric characterization of laser propagation

Summary

Victor-Cristian Palea

October 13, 2021

Contents

Introduction

Keywords: phase modulation, beam synthesis, paraxial wave equation, non-diffractive beams

This thesis' aim is to identify and introduce theoretical models that allow for the characterization and synthesis of optical profiles and/or beams that showcase novel properties during propagation. These properties can vary from complex amplitude-only distributions, to amplitude and phase profiles in a plane that is perpendicular to the propagation axis. The first type is associated with the phase retrieval problem[1] for which a solution is given by the Gerchberg-Saxton algorithm[2] (GSA). GSA uses two amplitude functions as inputs, one for the input and the other one for the desired output profile. A phase function is computed based on these two functions, for each one of them, thus creating two complex valued profiles. By propagating the input complex profile an amplitude similar to the desired one should be generated. In the case of GSA the propagation is given by a Fast Fourier Transform (FFT).

GSA also has limitations, with one of the most evident being the use of FFT to describe the propagation. Assuming the case where an actual Fourier Transform (FT) is applied instead of the FFT, the propagation in-between the input and the output should either be modeled such that it is given by FT, or an optical system has to be introduced that performs this operation e.g. a lens placed at distances equal to its focal length from both the input and the output. This issue has been generalized by Dorsch et al.[3] where the propagation is described by a Fresnel transform although this still implies that the optical system is restricted to the free space propagation operator only.

The solution of this problem has been proposed by Yang et al.[4] through the introduction of the Yang-Gu Algorithm (YGA) which:

- Generalizes the propagation through the use of an operator.
- The propagation operator can be unitary or not i.e. the second case is exemplified by the use of an aperture in the optical system.
- Generalizes the phase retrieval problem by introducing three additional case studies:
  - known input amplitude and output amplitude ⇒ compute input phase and output phase (similar to GSA)
  - known input phase and output phase ⇒ compute input amplitude and output amplitude
  - known input amplitude and output phase ⇒ compute input phase and output amplitude
  - known input phase and output amplitude ⇒ compute input amplitude and output phase

Using either the above mentioned algorithms, modified versions(e.g. for GSA[5, 6, 7, 8, 9, 10, 11]), or alternative ones (e.g. error-reduction algorithm[12], steepest-decent method[12], input-output algorithm[12], or Fresnel ping-pong algorithm[3]), the problem of synthesizing the amplitude of an optical profile can be optimized based on the particular case study of interest.

The case of synthesizing complex valued profiles however is not solvable via these algorithms since they impose either the amplitude or the phase to be known at both the input and output, but never both. This implies that one cannot choose a complex valued output profile. An optical profile that requires both amplitude and phase for the initial condition is the Airy beam[13].

The Airy beam gathered interest in optics due to its propagation related properties, namely it being non-diffractive[13, 14, 15], self-accelerated [13, 14, 15], and self-healing[16]. In the initial theoretical introduction by Berry et al.[13] the Airy function, used as an initial condition, was defined on the entire spatial domain and was not of class L_2. This made the physical implementation in optics of this theoretical result to be unusable unless truncation is considered, which further led to having the non-diffractive and self-accelerating properties valid only for a finite propagation distance and in an approximate manner.

From the point of view of synthesizing an Airy profile there are two main methods. The first one uses the truncation of the Airy beam. Due to it, after a finite propagation distance (forward or backward) the truncated Airy profile spreads out giving an amplitude profile similar to a Gaussian. Knowing the analytic solution for the propagation of a truncated Airy beam one can take the phase component at a given position where the amplitude resembles a Gaussian and apply it to modulate the phase of a Gaussian laser beam of similar width. The second method consists of using the trajectory of the global amplitude peak. It was proposed by Kaganovsky et al.[17] and uses the trajectory as the geometric space where a caustic is imposed. The profile is thus computed such that it generates that caustic, resulting a function that is similar to the Airy function.

Thus it can be concluded that these methods either make use of a theoretical result (Siviloglou et al.[14, 15]), or impose constraints that are related only to a part of the spatial domain (Kaganovsky et al.[17] and other results that follow from considering the trajectory of a caustic[18, 19, 20]). They do not however explain why the Airy function generates a beam with the above mentioned properties by considering only its shape and the propagation model. Also the non-diffractive and self-accelerating behaviors, which have been used in the context of proving that the Airy function is unique[21], have been expressed with a relation such as

$$A(x,z) = A(x - f(z)) \tag{I}$$

where A is the beam amplitude, x indicates the transverse axis, z the propagation one, and f is a continuous function. In the case of the Airy function $f(z) = az^2$. This way of introducing a constraint is non-trivial if truncation is not used, since none of these properties are valid unless an infinite domain is chosen. For the truncated case an approximation of the mathematical expression for the constraint has to be considered.

The shape related properties that can be used to describe the non-diffractive and self-accelerating properties are presented in chapter 1. Based on these properties an algorithm is proposed for constructing numeric initial condition which are then used for validation.

In chapter 2 the subject is shifted from the identified shape related properties to proposing a method of generalizing the condition $A(x,z) = A(x - f(z))$ by expressing it as a transport equation[22] thus expressing the constraint as a set of isotimic curves i.e. the function A^2 is constant on an isotimic curve. The validation consists of applying the isotimic curves model on non-diffractive scenarios with

curves given by polynomials of order upto 2 i.e. parabolic trajectories with controllable initial velocity and acceleration.

For experimental validation a method of synthesizing a complex optical profile with the desired properties is required. Chapter 3 covers this topic through the use of phase modulation using one or two phase masks. The aim here is to synthesize optical profiles with known amplitude and phase, unlike GSA or YGA where only one of them can be enforced.

Chapter 4 proposes a method of synthesizing optical beams from a given amplitude function that may of may not correspond to an actual beam. In the former case the method proposed allows for the computation of the phase function for an optical beam for which only the amplitude is known. The later case is the actual beam synthesis method since it starts from a desired yet not physical amplitude function and allows the computation of a physical complex optical beam with amplitude similar to the desired one. The method is tested for amplitude functions consisting of narrow Gaussian functions that are translated given a condition similar to $A(x, z) = A(x - f(z))$. This method is presented in the context of the work of Aborahama et al.[23].

In chapter 5 the numerical module that I have been developed to test, validate and implement all the above mentioned results, algorithms of models is presented. The module's evolution is given in a chronological manner. The main stages of development are presented using versions of the software, while also mentioning which version was used in order to validate parts of the theoretical and numerical results.

In the last chapter 6 the conclusions of the thesis are briefly enumerated.

Chapter 1

Local properties model - the qualitative approach

The properties that are identified in this chapter are based on an observation made on the solution of the paraxial wave equation[24]

$$\partial_t \psi = i\partial_x^2 \psi \tag{1.1}$$

where $t = z/2kx_0^2$ is a dimensionless coordinate corresponding to the propagation axis z, and $x = x/x_0$ corresponds to a dimensionless transverse axis.

Considering that the solution $\psi(x,t)$ can be expressed as a Taylor series around $t = 0$, then

$$\psi(x,t) = \sum_{n=0}^{\infty} \frac{t^n}{n!} \partial_t^n \psi(x,0). \tag{1.2}$$

Using the propagation equation (1.1) for (1.2) gives

$$\psi(x,t) = \sum_{n=0}^{\infty} \frac{(it)^n}{n!} \partial_x^{2n} \psi(x,0). \tag{1.3}$$

From (1.3) one can make the observation that if $\exists x_0$ such that $\partial_x^{2n}\psi(x_0,0) = 0, \forall n \geq 1$, then the solution remains constant at $x = x_0$. Additionally, for the general solution (1.3) if $t \ll 1$ then one can consider that the contribution of each term in (1.3) decreases with n, thus having only the first term to be the most relevant in describing the evolution of the solution.

These observations lead to the hypothesis of having a point x_0 for which $\partial_x^{2n}\psi(x_0,0) = 0, \forall 1 \leq n < k_0$, where $k_0 \in \mathbb{N}^*$, and $\partial_x^{2(k_0+1)}\psi(x_0,0) \neq 0$. In this situation the contribution to the variation of ψ at $x = x_0$ with respect to t is given mainly by the Taylor series term of order (k_0+1), which gives

$$\psi(x_0,t) \approx \psi(x_0,0) + \frac{(it)^{(k_0+1)}}{(k_0+1)!} \partial_x^{2(k_0+1)} \psi(x_0,0). \tag{1.4}$$

Equation (1.4) shows that the variation of the solution depends on k_0 and the partial derivative of even order. By analyzing the cases that result for various values of k_0 one can qualitatively deduce the

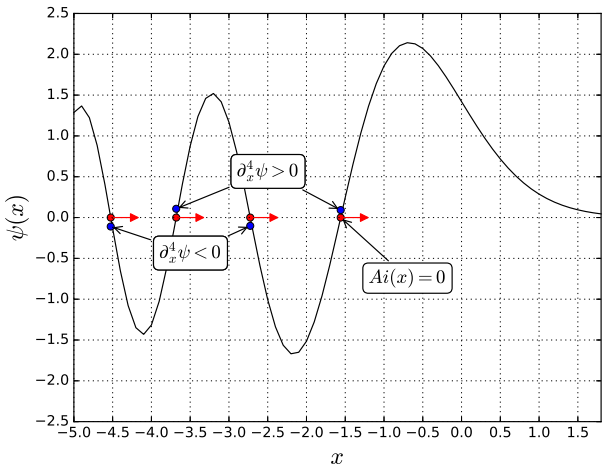


Figure 1.1: Plot of the Airy function. The red dots at positions x_i satisfy the conditions $\psi_0(x_i) = 0$ and $\partial_x^2\psi_0(x_i) = 0$, while the blue dots give the value of $\partial_x^4\psi_0(x_i)$. For all four cases, the shift is to the right on the spatial axis.

direction along which the solution shifts. This result is applied on the Airy function and the direction of shift correspond to the one given by the Airy beam (figura 1.1).

From this observation it can be implied that a qualitative method of characterizing the shift starts from identifying all the points x_0 for which $\partial_x^{2n}\psi(x_0, 0) = 0, \forall 1 \leq n < k_0$, meaning that they are at least inflection points. If all of them indicate a shift in the same direction then it should be expected that the entire function will shift globally for $t \ll 1$. The result has been investigated for all the relevant scenarios of k_0 (modulo 4) due to the term i^{k_0+1}, and all of them have been validated numerically for $k_0 > 1$ although the propagation distance on which the results are true decreases with k_0. Thus only the case where $k_0 = 1$ should be used.

This property has been used in the following to construct initial conditions. The method uses the above mentioned results but applies them on a discrete grid due to the numeric implementation.

The construction of initial conditions can be summarized as follows:

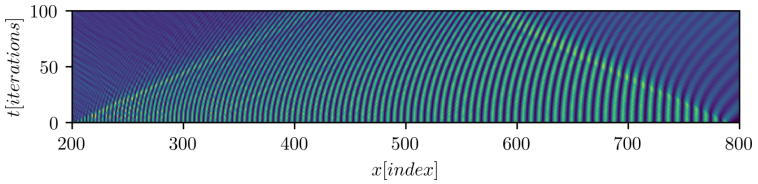
- Generate a set of points at which it will be enforced that $k_0 = 1$.
- Fill the intervals in-between each two neighboring points defined above with a function that does not generate additional inflection points.

I have chosen five types of set generators namely arithmetic series, geometric series, hyperbolic series, arithmetic series with random rate and geometric series with random rate. For the filler functions I have chosen parabola, sine, hyperbolic function, Bezier function, and random, where the later has been implemented by generating the second order derivative of the function using random numbers and then integrating numerically twice.

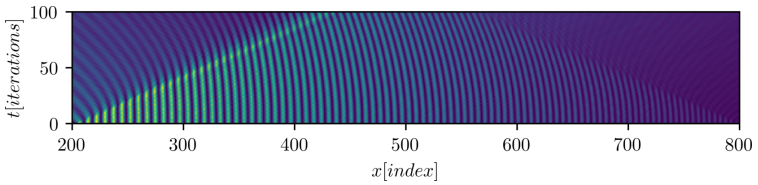
The resulting initial conditions have been created using by combining each set generator with each filler function. The aim has been to check if all do indeed propagate as described by the qualitative model.

Three of the investigated cases are represented in figure 1.2 where one can observe that the local amplitude peaks do propagate on a curved trajectory. For a qualitative analysis of the results numerical fitting has been applied to the local amplitude peaks using a second order polynomial. The acceleration coefficients have been retrieved for each local amplitude peak, the average of all their values has been computed for each case and the mean values are given in table 1.1.

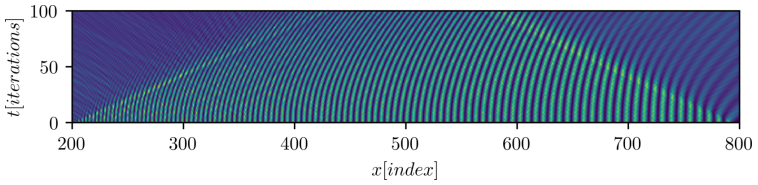
Thus one can conclude that by manipulating the position of the inflection points one can characterize the direction a profile shifts, while also allowing for the actual construction of optical profiles. This however did not allow for a quantitative way of controlling the acceleration term.



(a) Geometric series set generator with sine filler function.



(b) Hyperbolic set generator with parabolic filler function.



(c) Random geometric set generator with Bezier filler function.

Figure 1.2: Amplitude profiles for three scenarios for emphasizing different directions for the curved trajectory and one random set generator case.

Table 1.1: Mean values of parameter a_2 for the scenarios that have been considered.

Set Gen.	Parabolic	Sine	Bezier	Hyperbolic	Random
Arith.	$2.13 \cdot 10^{-3}$	$2.21 \cdot 10^{-3}$	$1.70 \cdot 10^{-3}$	$3.74 \cdot 10^{-4}$	$4.54 \cdot 10^{-4}$
Geom.	$5.21 \cdot 10^{-3}$	$5.63 \cdot 10^{-3}$	$6.37 \cdot 10^{-3}$	$5.21 \cdot 10^{-3}$	$1.23 \cdot 10^{-3}$
Hyper.	$-3.08 \cdot 10^{-3}$	$-2.55 \cdot 10^{-3}$	$-1.95 \cdot 10^{-3}$	$-3.08 \cdot 10^{-3}$	$-4.31 \cdot 10^{-3}$
R. Arith.	$4.71 \cdot 10^{-4}$	$8.57 \cdot 10^{-4}$	$9.48 \cdot 10^{-4}$	$4.90 \cdot 10^{-4}$	$7.72 \cdot 10^{-4}$
R. Geom.	$4.38 \cdot 10^{-3}$	$4.03 \cdot 10^{-3}$	$4.80 \cdot 10^{-3}$	$4.38 \cdot 10^{-3}$	$1.69 \cdot 10^{-3}$

Chapter 2

Isotimic curves model - The quantitative approach

The quantitative approach has been proposed based on an observation and limitation imposed by the truncation of the Airy beam. Without the truncation, the non-diffractive and self-accelerated propagation could have been described by

$$A(x,z) = A(x - az^2) \tag{2.1}$$

where A is the amplitude of the beam, while the non-diffractive and self-accelerated propagation are given by $x - az^2$. Using truncation however, the relation (2.3) is not applicable because the two properties are valid qualitatively only on a finite propagation domain.

In order to bypass the possible difficulties of describing some beam properties in a mathematical manner, an analogy between (2.1) and the transport equation[22] has been realized. For the partial differential equation

$$2az\partial_x f + \partial_z f = 0 \tag{2.2}$$

a solution is any function of the type $f(x - az^2)$. This result can be interpreted from a point of view that is close to the characteristics method. If one considers a set of isotimic curves Γ (on which the amplitude is constant) as a means of describing the amplitude of the optical beam, then these curves can be used to generate a transport equation.

Due to the fact that on each curve the amplitude is constant, it is then true that the tangent of the curves $\mathbf{T}_\Gamma$ is perpendicular to $\nabla|\psi|$ at each point. This result can be written as

$$\mathbf{T}_\Gamma \cdot \nabla|\psi| = 0. \tag{2.3}$$

For simplicity in the following I will consider only parametric curves of the type $\Gamma(s) = (g(s), h(s))$ where g and h are some arbitrary differentiable functions. It results from here that $\mathbf{T}_\Gamma = (\partial_s g, \partial_s h)$, which transforms (2.3) into

$$\partial_s g \partial_x|\psi| + \partial_s h \partial_z|\psi| = 0. \tag{2.4}$$

To equation (2.4) the propagation equation

h

Table 2.1: Comparison between the inputs (a_{in} and b_{in}) and fitting (a_{fit} and b_{fit}) values of parameters a and b. The values are expressed in the table.

a_{in}	-1.0	1.0	-1.0	1.0	-1.0	1.0	-1.0	1.0
a_{fit}	-0.99	0.99	-1.0	0.99	-1.0	1.0	-1.0	0.99
b_{in}	-3.0	-3.0	-2.2	-2.2	-1.4	-1.4	-0.6	-0.6
b_{fit}	-2.9	-2.96	-2.14	-2.18	-1.36	-1.41	-0.57	-0.58

$$\partial_z \psi = \frac{i}{2k} \partial_x^2 \psi. \tag{2.5}$$

can be added, where $k = 2\pi/\lambda$ is the wave number, in order to highlight the evolution of the amplitude squared function as

$$\partial_z |\psi|^2 = \frac{i}{2k} \left((\partial_x^2 \psi)\psi^* - \psi(\partial_x^2 \psi^*) \right). \tag{2.6}$$

Since the isotimic curves have the same shape for $|\psi|$ as for $|\psi|^2$, and since it is less cumbersome to use $|\psi|^2$ instead of just the amplitude, relation (2.4) becomes

$$\partial_s g \partial_x |\psi|^2 + \partial_s h \partial_z |\psi|^2 = 0. \tag{2.7}$$

If the curves are particularized even further to be of type $(g(z), z)$ and

$$g(z) = \sum_{n=0}^{\infty} \frac{z^n}{n!} \partial_z^n g(0),$$

then by introducing (2.6) in (2.7) one gets

$$\partial_x |\psi|^2 \left(\sum_{n=0}^{\infty} \frac{z^n}{n!} \partial_z^{n+1} g(0) \right) + \frac{i}{2k} \left((\partial_x^2 \psi)\psi^* - \psi(\partial_x^2 \psi^*) \right) = 0. \tag{2.8}$$

By evaluating equation (2.8) at $z = 0$ an ordinary differential equation in the x variable is obtained and the term g is reduced to only the term of order $n = 0$. Thus the interpretation of the resulting equation is that a differential equation is obtained which if solved returns an initial condition for which the direction of the isotimic curves is given by $(\partial_z g(0), 1)$.

For terms of orders $n > 0$ of the g series one has to apply ∂_z on (2.8), evaluate the resulting equation at $z = 0$ and then solve the differential equation.

The method allows for repeatedly applying ∂_z without an obvious constraint with the exception of the increasing difficulty of solving the resulting differential equation. However, based on the result of Unnikrishnan et al.[21], the Airy beam is the only non-diffractive beam for the propagation equation of interest. It is due to this reason that it should be expected that the isotimic curved method to work in the non-diffractive case only for curves given by polynomials of order upto 2. These cases have been investigated for various values of the coefficients corresponding to the initial velocity (labeled a) and acceleration terms (labeled b) of g. The results for the case where both coefficients are controlled at the same time are given in table 2.1. One should notice that the input values and the ones retrieved by fitting the local amplitude peak trajectories are similar.

An attempt at applying the method for polynomials of orders upto 4 has also been tried. The motivation for it, even though the theory limits the non-diffractive cases to order 2, is the existence of optical beams with a global amplitude peak that follows a trajectory more complex than a parabola[18, 20]. The numerical results in all cases beyond order 2 exploded after few iterations thus providing negative results.

The method has been considered only for non-diffractive scenarios, but an application of a diffractive Gaussian beam indicated that the analytic derivation of the isotimic curves from the solution and the method based on (2.8) returned the same coefficients of the Taylor series upto $n = 2$. This indicates that the method should be applicable on diffractive scenarios also.

Chapter 3

Amplitude and phase profile synthesis

Any general optical system can be formalized as in figure 3.1. Given the optical system and the input, the beam can be forward propagated in order to compute the output profile. The backward propagation is also possible, if the propagation model allows it, for computing the required input given an output profile and the optical system. This allows for splitting the optical system in order to include a controllable phase mask. In this scenario, the phase mask is computed by forward propagating the input through System 1 and backward propagating the output through System 2. At the phase mask plane the two resulting beams from the forward and backward propagation should match in both amplitude and phase in order to have an ideal reconstruction of the output profile. The match of the phase can be done using the phase mask. The match for the amplitude can be done using an amplitude mask, but we consider only the phase mask modulation scenario. This implies that any difference in the forward and backward propagated amplitude profiles should be handled by choosing an adequate optical system, labeled System 1, which thus has the purpose of preparing the amplitude profile of the input for the match to occur. The same is true for the later optical system, labeled System 2, since by an adequate choice of optical elements, it can help increase the amplitude match at the phase mask plane when the backward propagation is carried out.

The optical systems used in this chapter are composed of four types of optical elements: free space of a given length along the propagation axis, amplitude masks, phase masks and lenses. Although lenses are defined in the following as phase masks, thus rendering their distinction unnecessary, due to their common use I have considered that they should be defined as separate optical elements.

The propagation through free space is done by solving the propagation equation

$$\partial_z \psi = \frac{i}{2k}(\partial_x^2 \psi + \partial_y^2 \psi) \tag{3.1}$$

which can be reduced to only 1 dimension on the transverse domain by eliminating the term $\partial_y^2 \psi$. In order to solve (3.1) I have used a FT based solver[24].

The masks are applied to an optical profile by multiplication

$$\psi := M \cdot \psi.$$

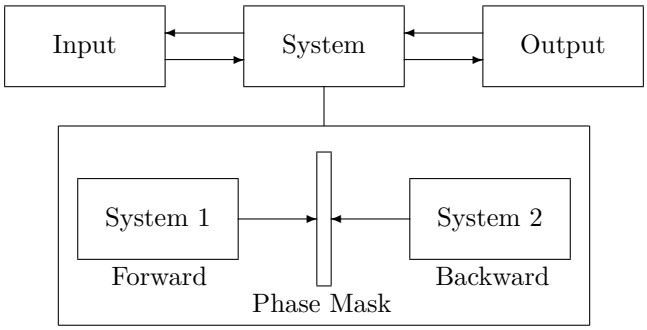


Figure 3.1: General optical system with input and output profiles. For introducing a controllable phase mask the optical system is split in 2 subsystems.

It should be mentioned that the amplitude masks should have real values inside the interval $[0, 1]$, while the phase masks should be expressed as $\exp(i\phi)$. For lenses the phase mask is computed using the focal length f and formula

$$M_l = \exp\left(ik\frac{x^2 + y^2}{2f}\right).$$

The propagation of a profile through an optical system thus becomes the sequential propagation through each optical element by applying its corresponding formula accordingly.

Having the optical elements and the propagation through an optical system defined, the algorithm for computing the phase mask is:

1. Define the input variables, namely the input profile ψ_{in}, the output profile ψ_{out}, and the optical subsystems S_1 and S_2.
2. Propagate backwards the output profile ψ_{out} through S_2 which gives the profile ψ_b.
3. Propagate forward the input profile ψ_{in} through S_1 which gives the profile ψ_f.
4. Calculate the phase mask with $M = \text{angle}(\psi_b/\psi_f)$.

One of the numeric results used for validation is depicted in figure 3.2. Subsystem 1 is empty or not used, thus having the modulation directly on the input profile. Subsystem 2 consists of a free space domain of 600mm. The input profile is a Gaussian and the output profile is a 2-dimensional Airy profile[14]. In order to quantitatively compare the similarity between the desired output profile and the one obtained with the retrieved phase mask, the cross-correlation of the two L^2 normalized functions is computed. The global maximum of the cross-correlation's amplitude is identified, and due to normalization, it should have a value inside the interval $[0, 1]$ thus allowing for the transformation of it into a percentage.

Diagram 3.2 describes two processes. During the first one the output profile is propagated backwards in order to get the amplitude on the phase mask plane. The input and backwards propagated profiles are compared in terms for amplitude. The width of the input Gaussian beam is set, assuming that we indeed use a Gaussian, such that the two amplitude functions are as similar as possible. Another method for obtaining an ever better amplitude match is to change the Gaussian width in order

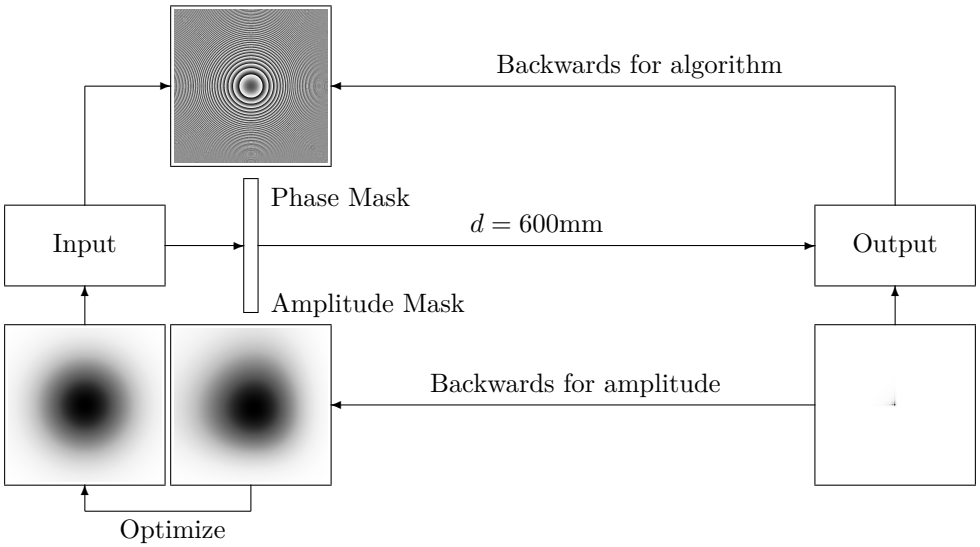


Figure 3.2: Diagram of the algorithm for optical System 1. The match between the input amplitude profile and the backwards propagated one (Backwards for amplitude line) is done by choosing a Gaussian with a similar width. The phase mask in this case is computed based on the backwards propagation through S_2 (Backwards for algorithm line).

to maximize the cross-correlation peak. After this optimization is done, the retrieved phase mask is the best result given the optical system we have considered, which in this case gave a similitude of 99.20%.

The algorithm has been validated experimentally using the system from figure 3.3. I have used a He-Ne tunable laser at a wavelength of 543nm with a standard deviation of the profile of 0.2mm, lenses with focal lengths and positions described in figure 3.3, a Holoeye spatial light modulator (SLM) model LC2002 with 800×600 pixels, 32μm pitch, and a 1SP CCD camera with a 3.2μm pitch.

The optimization for the optical setup has been made via the positioning of the lenses and the distances in-between them, which correspond to the numerical values from figure 3.3. The computed phase mask and the measured Airy intensity profile are also shown in figure 3.3. The scaling between the measured and theoretical amplitude profiles has been checked returning a difference of approximately 44μm between the main intensity peak and its neighbors on the horizontal and vertical axes. The trajectory of the main intensity peak has been retrieved by computing the coefficient corresponding to the acceleration of a parabola giving $0.119\mu m/mm^2$, value that is close to the one retrieved from the experimental data, namely $0.13\mu m/mm^2$ along the x axis and $0.136\mu m/mm^2$ along the y axis. The differences between measured and numerically computed data can be attributed to the amplitude mismatch at the plane of the phase mask and tolerances regarding the alignment of the lenses or measurement of the distances in-between them.

An extension of the algorithm consists of using two phase masks that are separated by an additional optical subsystem instead of just one mask. The algorithm for a system with two phase masks can be defined through the following steps:

1. Define the input variables, namely the input profile ψ_{in}, the output profile ψ_{out}, and the optical subsystems S_1, S_2 and S_3.

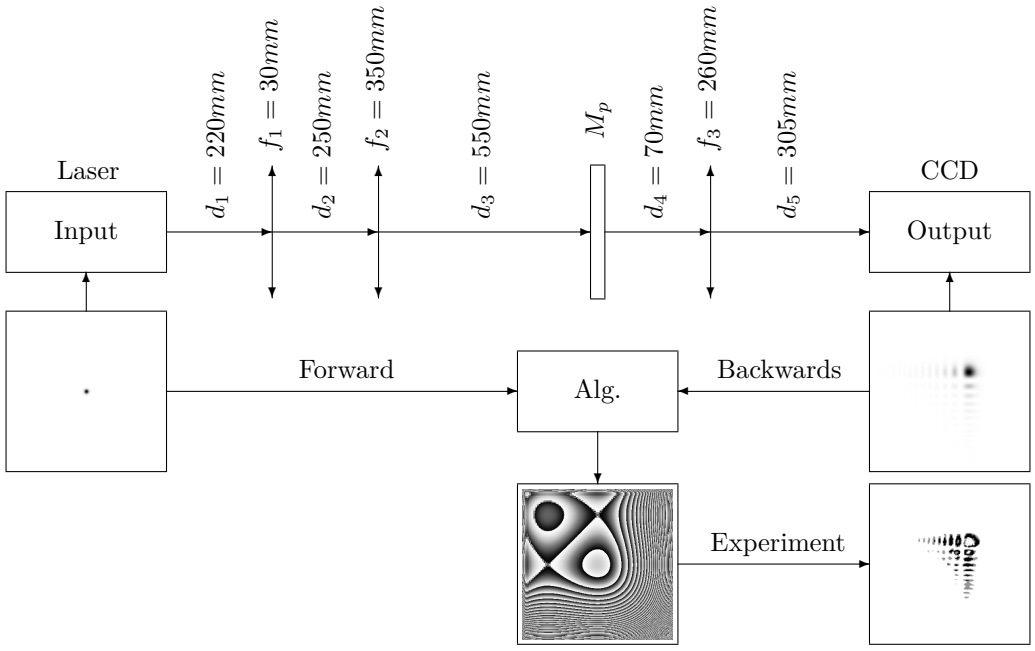


Figure 3.3: Experimental setup and procedure.

2. Propagate backwards the output profile ψ_{out} through S_3 which gives the amplitude profile at the plane of the second phase mask A.

3. Apply the one mask version of the algorithm for the optical system composed of S_1 and S_2 with ψ_{in} at the input and A at the output, which gives the first phase mask M_1.

4. Apply the one mask version of the algorithm for the optical system composed of $S_1 + M_1 + S_2$ and S_3 with ψ_{in} at the input and ψ_{out} at the output, which gives the second phase mask M_2.

A numerical scenario for validating the extended algorithm is given in figure 3.5 where a more complex profile consisting of a grid of dots, both thick and narrow circles and a square is to be synthesized. Subsystem 1 is empty thus M_1 is applied directly on the input profile. M_2 is positioned at $400mm$ from M_1 which defines subsystem S_2, while S_3 consists of a lens with focal length of $400mm$ positioned at $400mm$ from both $M_2$and the output plane. The phase masks M_1 and M_2 are computed using the extended algorithm and then are used to check the similarity between the initial output and the synthesized one. After optimization the similitude is of 92.66%. As a comparison, if we exclude M_1 and subsystem S_2 and use the original algorithm for the system that remains, the similarity is 70.17%, which indicates that the extended algorithm returned a better result.

Based on the above mentioned results, one can conclude that both algorithms can be used to synthesize optical beams when both amplitude and phase are defined by the user.

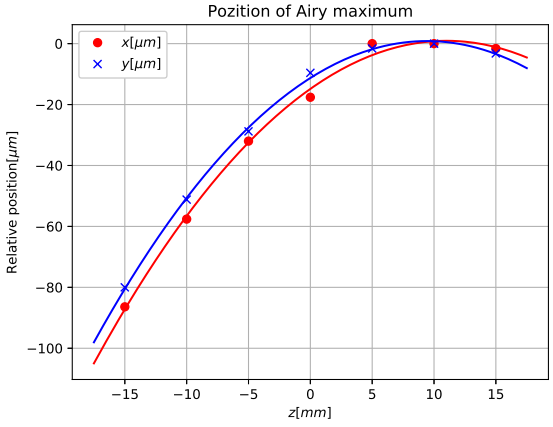


Figure 3.4: The measured and fitted trajectory of the main intensity peak for a 2-dimensional Airy beam along the x and y axes.

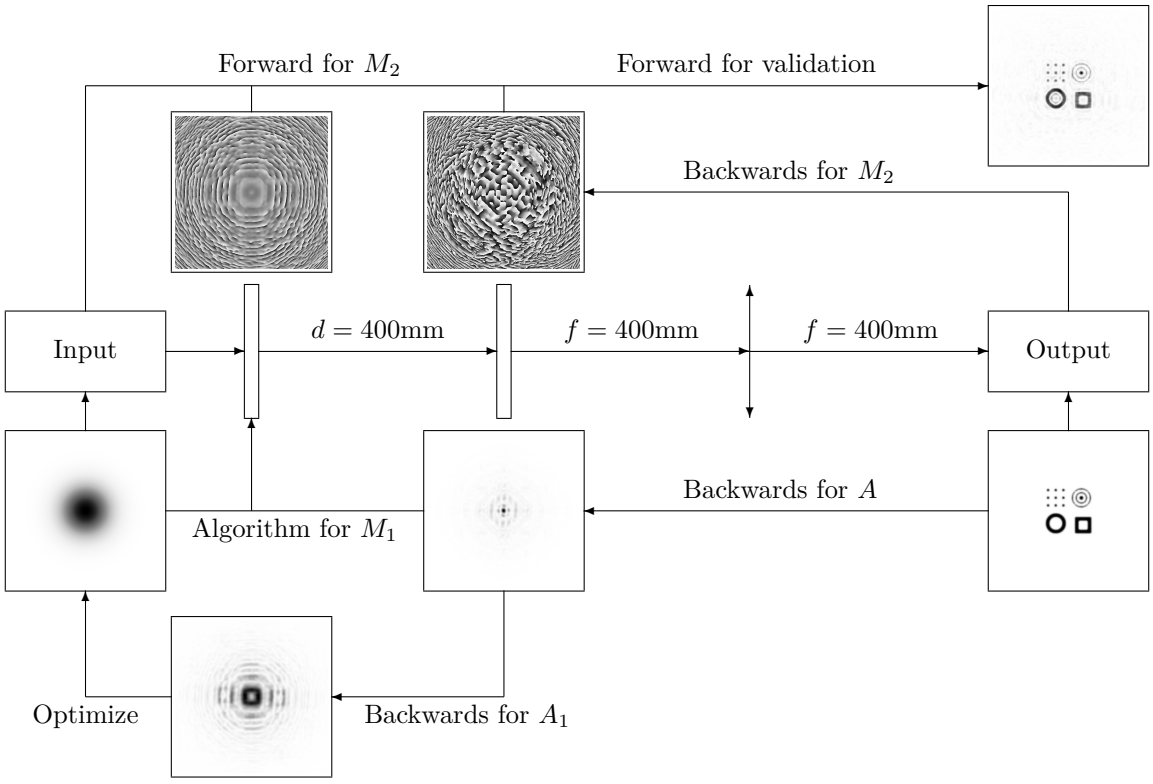


Figure 3.5: Diagram that indicates how the extended algorithm is applied on a system with two phase masks.

Chapter 4

Beam synthesis from an amplitude function

The approach of Aborahama et al.[23] can be considered to be the theoretical motivation for this chapter, namely to identify a method of computing a solution to the paraxial wave equation starting from an arbitrary function with the criterion that the two functions should resemble.

My approach uses a strictly positive function as a starting point for the amplitude of the solution, thus requiring the computation of the phase function. This is handled based on the ability of writing the phase in terms of the amplitude function

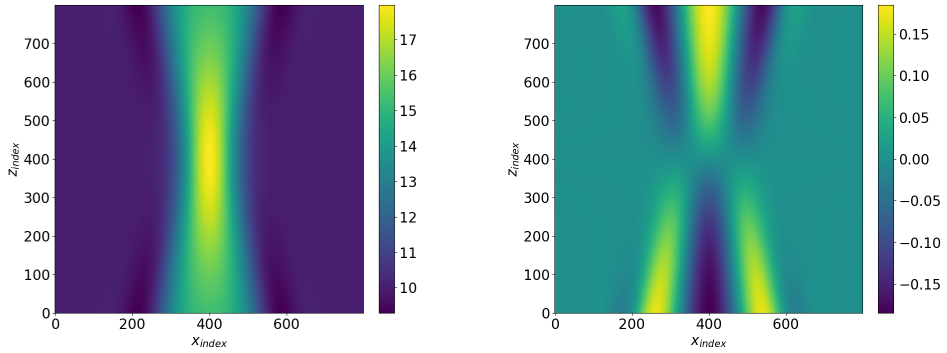
$$\nabla\phi = \mathbf{F} = \begin{pmatrix} -\frac{k}{\bar{A}^2}\partial_z \int_0^x \bar{A}^2 dx' \\ \frac{1}{2k\bar{A}}\left(\partial_x^2\bar{A} - \bar{A}\left(\frac{k}{\bar{A}^2}\partial_z \int_0^x \bar{A}^2 dx'\right)^2\right) \end{pmatrix}. \tag{4.1}$$

where $\bar{A}$ is the amplitude function. Equation (4.1) works better when the optical beam is given by the solution of interest ψ to which a constant that is greater than $\max(A)$ is added.

If $\bar{A}$ corresponds to the amplitude of an actual optical beam, then by solving equation (4.1) one gets the phase of the beam up to a constant at each position along the propagation axis.

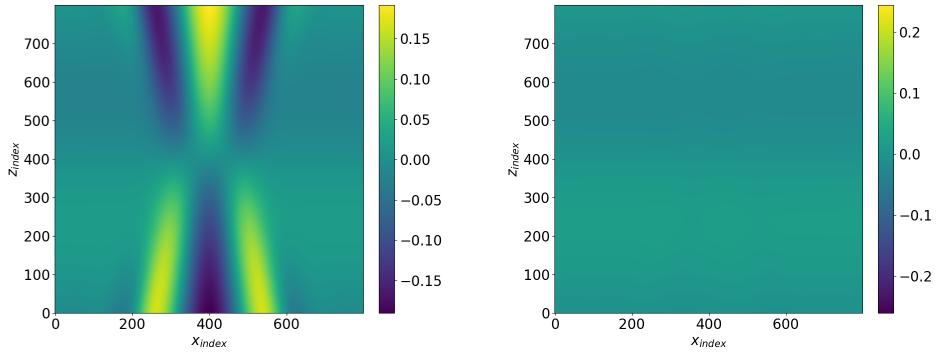
In order to solve equation (4.1) I have applied a divergence operator in order to obtain a Poisson equation, which is then solved using a FT based method and by writing the Laplace operator as a matrix using finite difference, thus transforming the problem into a standard linear system of equations.

Both methods have been testes initially on a well known optical beam, namely a numerical Gaussian beam to which a constant has been added. The results for the phase computation are shown in figure 4.1. By cmparing cases (c) and (e) wich (b) one can notice that the phases are comparable. The differences between the phases computed using (4.1) and the one from the numeric propagation also confirm the results of the retrieval.



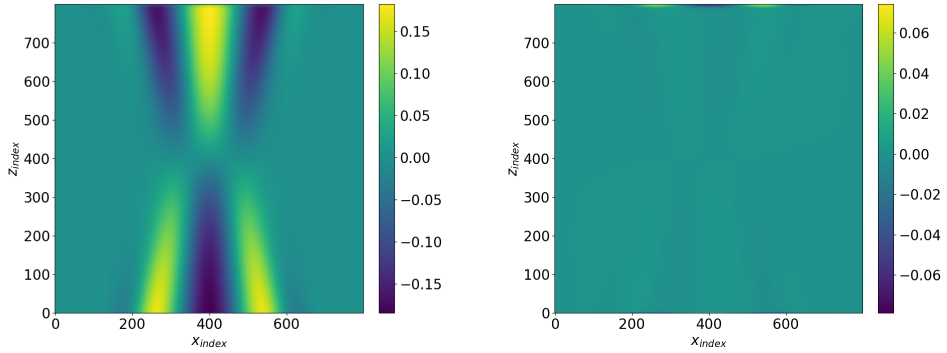
(a) Amplitude Gaussian beam.

(b) Phase Gaussian.



(c) Phase with FT.

(d) Phase difference for FT.

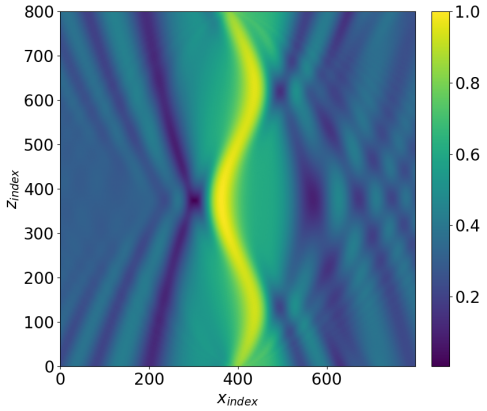


(e) Phase with LS.

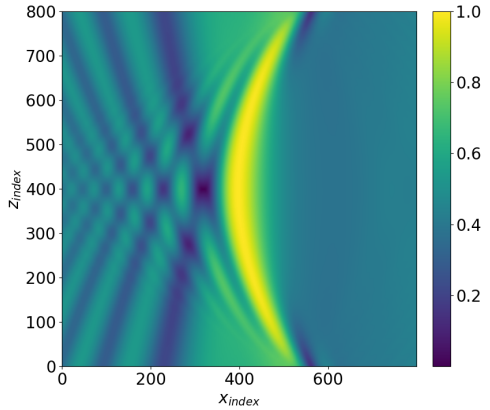
(f) Phase difference for LS.

Figure 4.1: The results of computing the phase using 4.1 from the amplitude of a Gaussian beam to which a constant has been added.

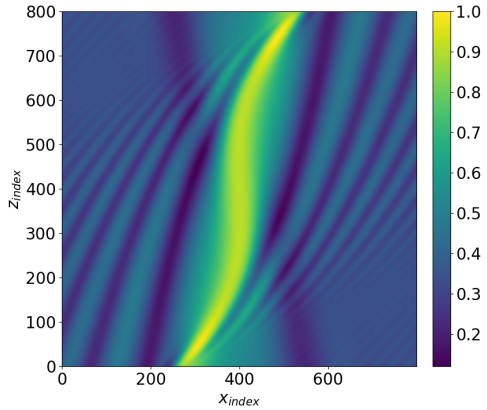
In the following I have used the same methods of computing the phase for amplitude that do not correspond to optical beams. The cases I have investigated resemble the ones from the caustics approach, in the sense that I have used a narrow Gaussian that is shifted such that it describes a desired trajectory. Using this method for constructing the amplitude functions, three cases have been considered: sine, second order polynomial and third order polynomial. Using the phase function computed for each case, the optical beams from figure 4.2 have been generated.



(a) Sine



(b) 2^{nd} order polynomial.



(c) 3^{rd} order polynomial.

Figure 4.2: Amplitude functions for the retrieved optical beams based on sine and polynomial trajectories.

Thus optical beams can be synthesized starting from a trajectory-like amplitude function, where as an intermediary step the computation of the phase function from amplitude data is required.

Chapter 5

Numerical solver

The development of the numerical solver began with the implementation of a finite differences scheme for solving the propagation equation. At this stage the implementation was not optimized to use efficient data structures and solvers. This however did not limit its ability to validate initial results that were relevant at the moment of its implementation. This is now considered to be **version 1.0**.

The shift to **version 2.0** represented the greatest jump in the evolution of the software due to the following major changes:

- The code was split in 4 classes:
  - numerical parameters
  - numerical solvers
  - initial condition generator functions
  - optical system simulators
- The introduction of numerical solvers for free space propagation using Crank-Nicolson scheme (with the option of using a *perfectly matched layer*[25]), non-linear Kerr term using forward Euler and Adams-Bashforth schemes, dispersion for the 1-dimensional transverse case using the *alternate direction implicit method* and FT based solver[24].
- The implementation of functions that generate initial conditions based on the inflection points and isotimic curves approaches.
- Linear and quadratic phase masks.
- Optical systems and the procedures for propagating a beam through them.
- Phase mask computation routines using the basic and extended algorithms.
- Tolerance analysis functions for misalignment or distance measurement errors.
- Calibration intended initial conditions such as grid of points, circles, squares, and a GSA implementation.
- Converter meant to switch from the pixel pitch used in the simulations to the pitch of the SLM used in the experimental setup.

- Functions for the identification and fitting of the local intensity peaks trajectories.

These changes have been implemented gradually based on the necessity at that moment. It is due to this way of implementing that all the results that are presented in the theses are generated using this implementation, with the only exception being the ones from chapter 4.

Versions 3.0 and **4.0** have been a cut-down, filtered and restructured **version 2**. The main reasons for deprecating parts of **version 2** have been the numerical optimization in terms of computational speed and the exclusion of non-essential code.

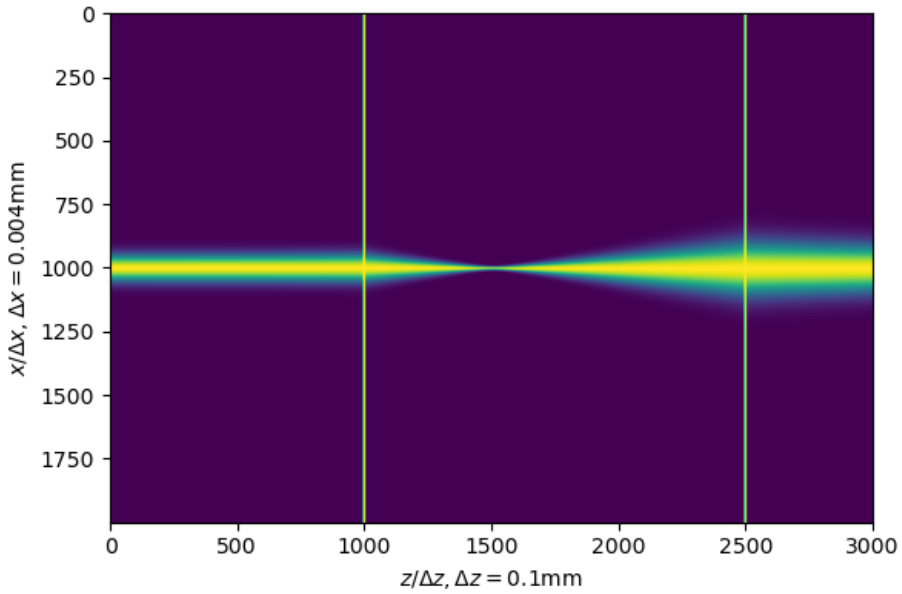


Figure 5.1: 1-dimensional Gaussian profile propagated through an optical system composed of two convergent lenses.

At the moment of writing this thesis **Version 5.0** is the stable version of the module. Due to the filtering from the previous two versions the PyParax module resulted along with a documentation and examples of code. This version is presented in greater detail in the extended version of the thesis and can be found at **https://github.com/victorcristianpalea/PyParax**.

This version uses only the FT solver due to its increased computational speed compared with Crank-Nicolson. This propagation solver is applied on optical systems, phase mask retrieval and tolerance analysis. A basic example for a result ofered by PyParax is showcased in figure 5.1. A Gaussian profile is propagated through an optical system made of two lenses. The distance between the input plane and the first lens is $100mm$, the one in-between the two lenses is $150mm$, and the distance between the last lens and the output plane is $50mm$. The focal lengths are $50mm$ for the first lens and $80mm$ for the second one. It can be seen in figure 5.1 that the beam is focused at approximately $50mm$ from

the first lens, and after passing through the second one the beam is slightly converging due to the focal length of $80mm$. It can be thus concluded that the optical profile propagates accordingly.

It should be mentioned that this version has been used in generating the results from chapter 4.

Versions 6.0 și 7.0, with the latter being still in development, are focused on restructuring the entire module again, so that the introduction of additional propagation models, optical media, numerical solvers and the modules inclusion into other projects is less cumbersome.

It can be concluded that the PyParax module has been motivated initially by the need to numerically validate the theoretical results presented in the previous chapters, for it to become gradually upgraded, allowing in the later stages for optical system prototyping and phase mask retrieval.

Chapter 6

Conclusions

This thesis can be summarized based on figure 6.1. Chapters 1 and 2 have focused on the construction of optical profiles from theoretical results. These profiles have to be synthesized using modulation, starting from a Gaussian beam. Chapter 3 covers this problem and describes the propagation through optical systems, introduces methods of computing phase masks, and hints at optimization possibilities regarding the design of an optical system. In chapter 4 a method of constructing beams for a given trajectory of the global amplitude peak using a phase retrieval method for the entire beam. Finally the numerical computation regarding all the above mentioned topics is covered in chapter 5 where the evolution of the PyParax module is presented.

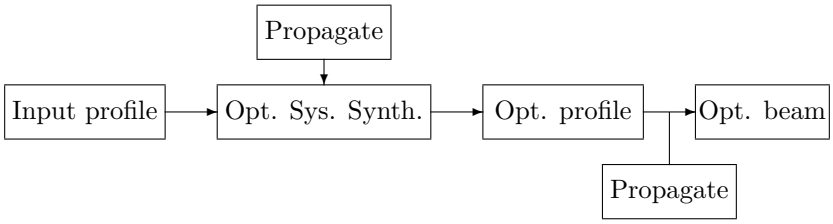


Figure 6.1: A general structure for the synthesis of optical beams. The input profile is passed through an optical system (Opt. Sys. Synth.) meant to synthesize an optical profile which propagated further can generate an optical beam of interest.

Personal contributions

My contribution to the subjects developed in this thesis covers everything that has been presented, namely:

1. the development of all the theoretical models and algorithms presented above.
2. the development of the PyParax module.
3. the planing, implementation and optimization of the experimental setup.

4. the acquisition of data, including curating, interpreting and processing.

5. writing the scientific articles, implying topic formulation, drafting, answering to reviewers.

The list of publications consists of 3 articles listed in the chronological order of their acceptance:

1. **Victor-Cristian Palea**, Liliana A. Preda. *The control of intensity peak dynamics for paraxial waves.* Journal of Engineering Mathematics, Vol. **115(1)**, No. p. 89-98, (2019).

2. **Victor-Cristian Palea**, Liliana A. Preda. *Isotimic curves for the description and control of intensity profile dynamics of solutions to the paraxial wave equation.* University POLITEHNICA of Bucharest Scientific Bulletin-Series A - Applied mathematics and physics. Vol. **81(2)**, No. p. 287-96, (2019).

3. **Victor-Cristian Palea**, Liliana A. Preda. *Construction of finite non-diffractive and self-accelerating laser beams.* Mathematical Methods in the Applied Sciences, 44: 11157- 11165, (2021).

Additionally I will mention one additional article that is has been in review at the moment of writing the thesis, but it has been accepted afterwards. This mention is due to the use of material from the manuscript in chapter 3.

1. **Victor-Cristian Palea**, Liliana A. Preda. *Forward-Backward propagation algorithm for phase mask design in the paraxial approximation* sent for review at Romanian Journal of Physics.

Selective bibliography

[1] L Taylor. The phase retrieval problem. *IEEE Transactions on Antennas and Propagation*, 29(2):386–391, 1981.

[2] Rw W Gerchberg. Phase determination for image and diffraction plane pictures in the electron microscope. *Optik (Stuttgart)*, 34:275, 1971.

[3] Rainer G Dorsch, Adolf W Lohmann, and Stefan Sinzinger. Fresnel ping-pong algorithm for two-plane computer-generated hologram display. *Applied optics*, 33(5):869–875, 1994.

[4] Guo-zhen Yang, Bi-zhen Dong, Ben-yuan Gu, Jie-yao Zhuang, and Okan K Ersoy. Gerchberg–saxton and yang–gu algorithms for phase retrieval in a nonunitary transform system: a comparison. *Applied optics*, 33(2):209–218, 1994.

[5] Tieyu Zhao and Yingying Chi. Modified gerchberg–saxton (gs) algorithm and its application. *Entropy*, 22(12):1354, 2020.

[6] Chien-Yu Chen, Wu-Chun Li, Hsuan-Ting Chang, Chih-Hao Chuang, and Tsung-Jan Chang. 3-d modified gerchberg–saxton algorithm developed for panoramic computer-generated phase-only holographic display. *JOSA B*, 34(5):B42–B48, 2017.

[7] Hone-Ene Hwang, Hsuan T Chang, and Wen-Nung Lie. Fast double-phase retrieval in fresnel domain using modified gerchberg-saxton algorithm for lensless optical security systems. *Optics express*, 17(16):13700–13710, 2009.

[8] Mona Mihailescu, Liliana Preda, Al M Preda, and EI Scarlat. Modified gerchberg-saxton algorithm for diffractive optical element image retrieval. *University" Politehnica" of Bucharest Scientific Bulletin, Series A: Applied Mathematics and Physics*, 67(4):65–76, 2005.

[9] Haichao Wang, Weirui Yue, Qiang Song, Jingdan Liu, and Guohai Situ. A hybrid gerchberg–saxton-like algorithm for doe and cgh calculation. *Optics and Lasers in Engineering*, 89:109–115, 2017.

[10] Gavin Sinclair, Jonathan Leach, Pamela Jordan, Graham Gibson, Eric Yao, Zsolt John Laczik, Miles J Padgett, and Johannes Courtial. Interactive application in holographic optical tweezers of a multi-plane gerchberg-saxton algorithm for three-dimensional light shaping. *Optics Express*, 12(8):1665–1670, 2004.

[11] AP Porfirev. Modification of the gerchberg-saxton algorithm for the generation of specle-reduced intensity distributions of micrometer and submicrometer dimensions. *Optik*, 195:163163, 2019.

[12] James R Fienup. Phase retrieval algorithms: a comparison. *Applied optics*, 21(15):2758–2769, 1982.

[13] Michael V Berry and Nandor L Balazs. Nonspreading wave packets. *American Journal of Physics*, 47(3):264–267, 1979.

[14] GA Siviloglou, J Broky, Aristide Dogariu, and DN Christodoulides. Observation of accelerating airy beams. *Physical Review Letters*, 99(21):213901, 2007.

[15] Georgios A Siviloglou and Demetrios N Christodoulides. Accelerating finite energy airy beams. *Optics letters*, 32(8):979–981, 2007.

[16] John Broky, Georgios A Siviloglou, Aristide Dogariu, and Demetrios N Christodoulides. Self-healing properties of optical airy beams. *Optics express*, 16(17):12880–12891, 2008.

[17] Y. Kaganovsky and E. Heyman. Wave analysis of airy beams. *Opt. Express*, 18(8):8440–8452, Apr 2010.

[18] Timor Melamed and Amir Shlivinski. Practical algorithm for custom-made caustic beams. *Opt. Lett.*, 42(13):2499–2502, Jul 2017.

[19] Gabriel Lasry, Timor Melamed, and Yaniv Brick. Manipulation and control of 3-d caustic beams over an arbitrary trajectory. *Opt. Express*, 28(14):20645–20659, Jul 2020.

[20] Yanping Lan, Fangrong Hu, and Yixian Qian. Generation of spirally accelerating optical beams. *Opt. Lett.*, 44(8):1968–1971, Apr 2019.

[21] K. Unnikrishnan and A. R. P. Rau. Uniqueness of the airy packet in quantum mechanics. *American Journal of Physics*, 64(8):1034–1035, August 1996.

[22] Lawrence Evans. *Partial Differential Equations*. American Mathematical Society, March 2010.

[23] Yousuf Aborahama, Ahmed H. Dorrah, and Mo Mojahedi. Designing the phase and amplitude of scalar optical fields in three dimensions. *Opt. Express*, 28(17):24721–24730, Aug 2020.

[24] A. Couairon, E. Brambilla, T. Corti, D. Majus, O. de J. Ramírez-Góngora, and M. Kolesik. Practitioner's guide to laser pulse propagation models and simulation. *The European Physical Journal Special Topics*, 199(1):5–76, November 2011.

[25] Chunxiong Zheng. A perfectly matched layer approach to the nonlinear schrödinger wave equations. *Journal of Computational Physics*, 227(1):537–556, 2007.