Investigating the Paraxial Wave Equation Using Green's Functions

Cole Martin

September 6, 2024

This paper was written under the direction of my advisors, Professor Reza Malek-Madani (USNA) and Professor Nick Moore (Colgate), and serves as my honors thesis.

1 Introduction

Our problem is to study laser beam propagation in a turbulent medium. Lasers are found across a huge range of real-world applications, and the Navy in particular is interested in lasers for use in communications and directed energy weaponry. We will study the behavior of laser beam propagation by investigating solutions to the Paraxial Wave Equation (PWE).

When the laser beam is propagating in a vacuum, the equation reduces to what is called the homogeneous form of the PWE, a partial differential equation with constant coefficients. The homogeneous PWE lends itself to computing exact solutions. Because of the simplicity of the homogeneous PWE, we have access to several exact and closed-form solutions to this equation. We note here that we will use homogeneous in reference to a medium of laser propagation which is the same everywhere and so the index of refraction is constant, as opposed to in reference to a homogeneous ordinary differential equation in which the trivial solution is zero. Consequently, nonhomogeneous will refer to a medium of propagation with a variable index of refraction.

Our goal, however, is to study the nonhomogeneous PWE, a partial differential equation with varying coefficients, for which we do not expect to find exact solutions. To that end, we intend to apply numerical methods to obtain approximate solutions. One of the main tools we will use is applying techniques from Laplace and Fourier transforms to obtain approximate solutions. We will use the exact solutions of the homogeneous PWE as benchmarks to calibrate the numerical method we will use later for the nonhomogeneous PWE.

The plan for the paper is as follows: Section 2 introduces the Paraxial Wave Equation, its physical parameters, and its derivation. Section 3 concerns the connection between the Heat Equation and the PWE, which is the motivation behind our numerical approach. Section 4 concerns the the concept of the Green's Function for the PWE. Section 4 presents a set of exact solutions of the PWE for a special set of initial conditions. Section 6 deals with the main numerical calculations applied to the homogeneous PWE. Section 7 is dedicated to the numerical solutions of the nonhomogeneous PWE. In Section 8 we discuss the current state of laser communications in underwater environments. Section 9 contains a conclusion and a summary of the paper. Appendices that follow contain the MATLAB codes.

2 The Paraxial Wave Equation

We will focus on the Paraxial Wave Equation, a partial differential equation which models the propagation of laser beams,

$$\Delta u + 2ik\frac{\partial u}{\partial z} + k^2\left(\frac{n^2}{n_0^2} - 1\right)u = 0. \tag{1}$$

As will become clear in this next section, in equation (1) u, which is a complex-valued function, is any component of either the electric or the magnetic field. The operator Δ in (1) is the usual Laplacian in x and y, which is sometimes denoted by $\Delta_\perp$. The distinguishing feature of a laser beam is its propagation axis, which is the z axis in (1). In addition, k is the light's wave number, that is, $k = \frac{2\pi}{\lambda}$, where λ is the wavelength of the light. In our numerical calculations, we will only consider red laser beams for which $\lambda = 633$ nanometers, which results in k having a value around 10^7. The term $(\frac{n^2}{n_0^2} - 1)$ represents the index of refraction. When the laser beam is propagating in a vacuum, the refractive index is constant, $n = n_0$, and the equation reduces to the homogeneous form.

Note that the equation in (1) is independent of time, signifying that the solution we are seeking has reached a steady-state. It is worth emphasizing that the independent variables x and y are distinguished from z and, as it will become clear later in our development, z plays the role of "time" as we interpret the solution u as a beam propagating in the z-direction.

In what follows, Equation (1) is supplemented by the initial condition

$$u(x, y, 0) = u_0(x, y), \quad x, y \in \mathbb{R}. \tag{2}$$

2.1 Derivation of the Paraxial Wave Equation

Light can be described as an electromagnetic wave propagating at the speed of light c through either a free-space vacuum (homogeneous) or a specific medium like air or water (nonhomogeneous). For a medium, the index of refraction $n(\mathbf{r})$ is defined as,

$$n(\mathbf{r}) = \sqrt{\frac{\epsilon(\mathbf{r})}{\epsilon_0}}$$

where ϵ is the permittivity of the medium and ϵ_0 is the permittivity of a reference homogeneous material. ϵ can vary in x, y, and z, and permittivity is the ability of a substance to store energy in an electric field. As an example, a perfect vacuum has a relative permittivity $(\frac{\epsilon}{\epsilon_0})$ of exactly one, while air has a relative permittivity of 1.0006 at standard temperature and pressure. One of the motivating applications we have in mind is how a laser beam propagates underwater. A maritime environment will cause significant variation in permittivity and in turn in the index of refraction.

Additionally, the permeability of a substance is its ability to store energy in a magnetic field, and will be denoted by μ. With both permittivity and permeability established, note that in our case,

$$c = \frac{1}{\sqrt{\mu\epsilon_0}}.$$

We will begin with Maxwell's Equations, which are a set of coupled partial differential equations,

$$\nabla \times \mathbf{E} = -\mu\frac{\partial \mathbf{H}}{\partial t}, \quad \nabla \times \mathbf{H} = \epsilon\frac{\partial \mathbf{E}}{\partial t},$$

where E and H are the electric and magnetic fields respectively. Taking the curl of the first equation and using the identity $\nabla \times \nabla \times \mathbf{E} = \nabla(\nabla \cdot \mathbf{E}) - \Delta\mathbf{E}$,

$$\nabla(\nabla \cdot \mathbf{E}) - \Delta\mathbf{E} = -\mu(\nabla \times \frac{\partial \mathbf{H}}{\partial t}).$$

Then taking the partial derivative in time of the second equation and combining,

$$\nabla(\nabla \cdot \mathbf{E}) - \Delta \mathbf{E} = -\mu \frac{\partial}{\partial t}(\epsilon \frac{\partial \mathbf{E}}{\partial t}).$$

Next, $0 = \nabla \cdot (\epsilon \mathbf{E}) = \nabla \epsilon \cdot \mathbf{E} + \epsilon \nabla \cdot \mathbf{E}$, so,

$$\nabla(\nabla \cdot \mathbf{E}) = -\nabla(\mathbf{E} \cdot \nabla(\ln \epsilon)),$$

and therefore

$$\mu \frac{\partial}{\partial t}(\epsilon \frac{\partial \mathbf{E}}{\partial t}) = \Delta \mathbf{E} + \nabla(\mathbf{E} \cdot \nabla(\ln \epsilon)).$$

This is known as the Generalized Wave Equation. Next, a series of five assumptions will help reduce the Generalized Wave Equation to the Paraxial Wave Equation. These assumptions are based on physical scale considerations specifically formulated for laser beam propagation (see [AP05], Chapter 4, and [Str78], Chapter 2):

1. ϵ is the permittivity within a given medium. We will assume that the permittivity is a function of x, y, and z, but not of time. Since ϵ is independent of time and the refractive index within a given medium $n(r) = \sqrt{\frac{\epsilon(x,y,z)}{\epsilon_0}}$,

$$\frac{n^2}{c^2} \frac{\partial^2 \mathbf{E}}{\partial t^2} = \Delta \mathbf{E} + \nabla(\mathbf{E} \cdot \nabla(\ln \epsilon)).$$

2. We will assume that the term $\nabla(\mathbf{E} \cdot \nabla(\ln \epsilon))$ is negligible relative to the two other terms above, and so will ignore it, to get,

$$\frac{n^2}{c^2} \frac{\partial^2 \mathbf{E}}{\partial t^2} = \Delta \mathbf{E}.$$

   Note that the magnetic field $\mathbf{H}$ satisfies the same equation, that is,

$$\frac{n^2}{c^2} \frac{\partial^2 \mathbf{H}}{\partial t^2} = \Delta \mathbf{H}.$$

   The above equations form 6 equations in the 6 unknowns, namely, the components of $\mathbf{E}$ and $\mathbf{H}$. These equations are uncoupled, so each component of the $\mathbf{E}$ and $\mathbf{H}$ satisfy the scalar wave equation

$$\frac{n^2}{c^2} \frac{\partial^2 u}{\partial t^2} = \Delta u, \tag{3}$$

   where u stands for any of the six components of $\mathbf{E}$ or $\mathbf{H}$.

3. We will assume that the wave is monochromatic, meaning that each component of the $\mathbf{E}$ (as well as $\mathbf{H}$) has the simple exponential dependence in time:

$$\mathbf{E}_x(x, y, z, t) = \psi(x, y, z) e^{-i\omega t}.$$

   After substituting the above expression in (3), we find that ψ satisfies the Helmholtz Equation

$$\Delta \psi + \frac{\omega^2 n^2}{c^2} \psi = 0. \tag{4}$$

4. Since we are interested in laser beam propagation, we will assume that the laser light is travelling in a beam form, that is, ψ has the special form

$$\psi(x, y, z) = u(x, y, z) e^{ikz}. \tag{5}$$

   The form we have selected, which is sometimes referred to as a magnitude-modulated solution, guarantees that the z direction is distinguished from the x and y directions, or in other words, the behavior of the solution in the z direction is distinctly different from its behavior in its cross-section, a feature that defines beam-like behavior in our model.

Substituting (5) in (4) results in

$$\Delta u + 2ik\frac{\partial u}{\partial z} + (\frac{n^2}{c^2}\omega^2 - k^2)u = 0. \tag{6}$$

Following common convention in physics, we substitute in $\omega = \frac{kc}{n_0}$ and $k = \frac{2\pi n_0}{\lambda_{vac}}$ in (6) and get

$$\Delta u + 2ik\frac{\partial u}{\partial z} + k^2(\frac{n^2}{n_0^2} - 1)u = 0. \tag{7}$$

5. Our final assumption is the so-called "paraxial" assumption. Comparing the magnitudes of $\frac{\partial^2 u}{\partial z^2}$ and $2ik\frac{\partial u}{\partial z}$ in (7), at least as a back-of-the-envelope calculation, we observe that $|\frac{\partial^2 u}{\partial z^2}| \ll |2ik\frac{u}{\partial z}|$. Therefore, $\frac{\partial^2 u}{\partial z^2}$ is ignored in deference to the $2ik\frac{u}{\partial z}$ term, resulting in the Paraxial Wave equation

$$\Delta_\perp u + 2ik\frac{\partial u}{\partial z} + k^2(\frac{n^2}{n_0^2} - 1)u = 0, \tag{8}$$

where $\Delta_\perp$ stands for the laplacian in x and y only.

This is the equation which we will be analyzing in the remainder of this paper, subject to the initial conditions

$$u(x, y, 0) = u_0(x, y), \tag{9}$$

where the function u_0 in (9) is typically a Gaussian function representing the structure of the laser beam at the aperture, $z = 0$. Our goal is to determine u for $z > 0$ under suitable conditions on k and n. In the next few sections we will present exact and numerical solutions of (8)-(9) in the homogeneous case (when $n = n_0$) and numerical solutions of (8)-(9) in the nonhomogeneous case when n varies spatially. To motivate our numerical approach, we first discuss this approach in the case of the Heat Equation.

3 Connection to the Heat Equation

As stated earlier, our approach to finding solutions of (8)-(9) will involve using the Green's function of the differential operator in (8). Because (8), at least in form, closely resembles the Heat Equation, we will begin with the latter, find its Green's function, which is also called the Heat Kernel, and test our numerical method, before returning to PWE in (8).

3.1 Derivation of Heat Kernel

We begin with the Heat Equation in one dimension, with diffusion coefficient c^2 and initial condition ϕ,

$$u_t - c^2 u_{xx} = 0, \quad u(x, 0) = \phi(x), \quad x \in \mathbb{R}. \tag{10}$$

Recall the definitions of the Laplace and Fourier transforms:

$$\mathscr{L}[f](s) = \int_0^\infty f(t)e^{-st}\,dt \qquad \mathscr{F}[f](\omega) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^\infty f(x)e^{ix\omega}\,dx. \tag{11}$$

Let U stand for the combined transforms of u, that is, U represents the Laplace transform of u followed by the Fourier transform of the resulting function in x.

$$U(s, \omega) = \mathscr{L}[\mathscr{F}[u(x, t)]] = \mathscr{F}[\mathscr{L}[u(x, t)]], \tag{12}$$

where the last equality follows since the two transform operations commute. Returning to (10), we take the combined transform of both sides of this equation:

$$\mathscr{F}[\mathscr{L}[u_t]] - c^2\mathscr{L}[\mathscr{F}[u_{xx}]] = 0. \tag{13}$$

Since $\mathscr{L}[u_t] = -u(x,0) + s\mathscr{L}[u]$, the first term on the left-side of (13) becomes $-\mathscr{F}[\phi(x)] + sU(s,\omega)$. Similarly, since $\mathscr{F}[u_{xx}]] = -\omega^2\mathscr{F}[u]$, the second term in (13) reduces to $-c^2\omega^2 U(s,\omega)$, and (13) is equivalent to the algebraic equation

$$-\mathscr{F}[\phi(x)] + sU(s,\omega) = -c^2\omega^2 U(s,\omega),$$

or

$$U(s,\omega) = \frac{1}{s + c^2\omega^2}\mathscr{F}[\phi(x)].$$

from which we obtain the solution $u(x,t)$ by inverting the two transforms:

$$u(x,t) = \mathscr{F}^{-1}[\mathscr{L}^{-1}[\frac{1}{s + c^2\omega^2}]\mathscr{F}[\phi(x)]]$$

Since $\mathscr{L}^{-1}[\frac{1}{s+c^2\omega^2}] = e^{-c^2\omega^2 t}$, we have

$$u(x,t) = \mathscr{F}^{-1}[\mathscr{F}[\phi(x)]e^{-c^2\omega^2 t}].$$

Finally, we exploit the fact that the Fourier transform of the convolution of two functions is equal to the product of the individual Fourier transforms of the functions, that is, $\mathscr{F}[f * g] = \mathscr{F}[f]\mathscr{F}[g]$, and the fact that $\mathscr{F}^{-1}[e^{-c^2\omega^2 t}] = \frac{1}{\sqrt{2\pi t}}e^{-\frac{c^2x^2}{t}}$, to arrive at the solution u as a convolution of the initial data ϕ with the kernel $\frac{1}{\sqrt{4\pi t}}e^{-\frac{c^2x^2}{t}}$:

$$u(x,t) = \phi(x) * \frac{1}{\sqrt{4\pi t}}e^{\frac{-c^2x^2}{t}}. \tag{14}$$

This is the solution to our Initial Value Problem defined in (10). We will let $G(x,t)$ denote the kernel in (14) and refer to it as the Heat Kernel or the Heat Equation's Green Function. We also note that the expression in (14) is equivalent to

$$u(x,t) = \int_{-\infty}^{\infty} G(x-\xi,t)\phi(\xi)\,d\xi. \tag{15}$$

3.2 Heat Kernel in $\mathbb{R}^d$

The technique described in the previous section can be generalized in a straightforward manner and apply to the Heat Equation, $u_t - \Delta u = 0$, in any dimension d. The Heat Kernel then takes the form

$$G(\vec{x},t) = \frac{1}{(4\pi t)^{\frac{d}{2}}}e^{\frac{-|\vec{x}|^2}{t}}, \tag{16}$$

and similar to (15), the solution to the Heat Equation in $\mathbb{R}^d$ is again the convolution of the Heat Kernel G and the initial condition $u(\vec{x},0) = \phi(\vec{x})$

$$u(\vec{x},t) = \int_{\mathbb{R}^d} G(\vec{x}-\vec{y},t)\phi(\vec{y})\,d\vec{y}. \tag{17}$$

This is therefore our solution to the Heat Equation in any dimension d. See [GS90] for more information.

3.3 Trapezoid Rule

We have many different options to numerically evaluate the integral equation in (17), including the Left/Right Hand Rule, the Trapezoid Rule, or adaptive quadrature methods such as MATLAB's "integral" function or Mathematica's "NIntegrate" function. We have experimented with these three main types of numerical integration, but will mainly focus on using the trapezoid quadrature method in this project.

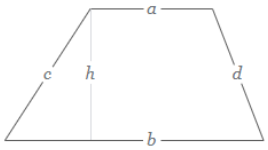

Figure 1: A basic example of a trapezoid.

The area of the above trapezoid is:

$$A = (\frac{a+b}{2})h.$$

Now consider a simple example, where $y = x^2 + 1$.

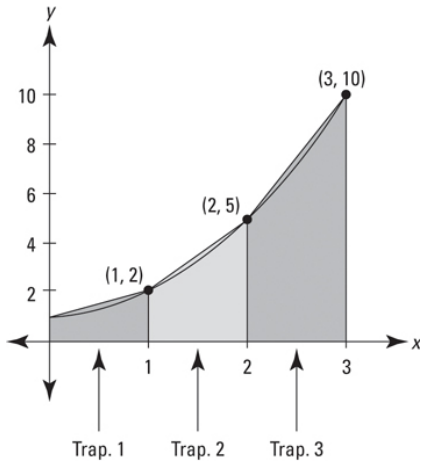

Figure 2: The function $y = x^2 + 1$, with the integral on [0,3] approximated by three trapezoids.

If our goal is to approximate $\int_0^3 f(x)\,dx$, then we can apply three trapezoids in order to calculate

$$\int_0^3 f(x)\,dx \approx \frac{1}{2}[f(0) + 2f(1) + 2f(2) + f(3)] = 12.5.$$

The exact value of the integral is 12, and so we are within 0.5 of the correct answer with only three trapezoids. Advantages of using this method are:

- Fast computation time.
- Accurate and reliable results (more accurate than Left/Right Hand Rules).
- Very easy to test and diagnose errors (easier than black box adaptive quadrature functions such as those mentioned above).

3.4 Heat Equation - Numerical Results

3.4.1 Example 1

Consider the Heat Equation in one dimension with a Gaussian initial condition,

$$u_t - u_{xx} = 0, \quad u(x,0) = \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}}. \tag{18}$$

We have created a MATLAB script which approximates the solution at any given time value using the integral equation above in (17). The integral is computed numerically using the trapezoid quadrature rule with 1000 trapezoids over the interval (-20,20).

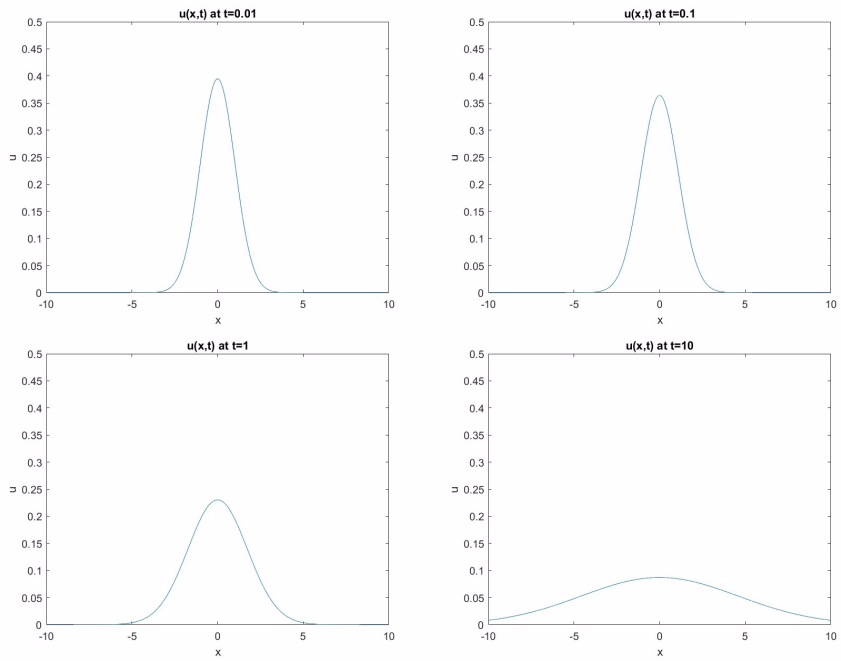

Figure 3: Solutions for Example 1 at different time values. Solutions are calculated numerically using the Heat Kernel and the trapezoid rule for $t = [0.01, 0.1, 1, 10]$.

The exact solution for this initial condition is,

$$u(x,t) = \frac{1}{2\sqrt{\pi(t+\frac{1}{2})}} e^{\frac{-x^2}{2+4t}} \tag{19}$$

The largest error between our numerical method and this exact solution at any of the above time values is $5.55 \cdot 10^{-15}$. The error increases towards the center of the Gaussian peak, but is negligible overall.

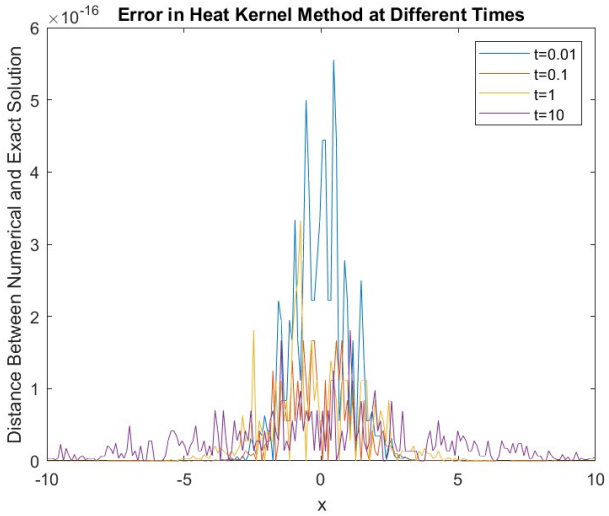

Figure 4: Error in numerical method for Example 1. Figure shows the absolute error as compared to the exact solution for the Heat Equation. These are compared at the same time values as in Figure 3.

3.4.2 Example 2

Now consider the Heat Equation in one dimension, but now with an initial condition of two superimposed Gaussian functions with different positions, amplitude, and variance.

$$u_t - u_{xx} = 0, \quad u(x,0) = \frac{3}{\sqrt{2\pi}} e^{\frac{-(x-1)^2}{2}} + \frac{2}{\sqrt{4\pi}} e^{\frac{-(x+2)^2}{4}}. \tag{20}$$

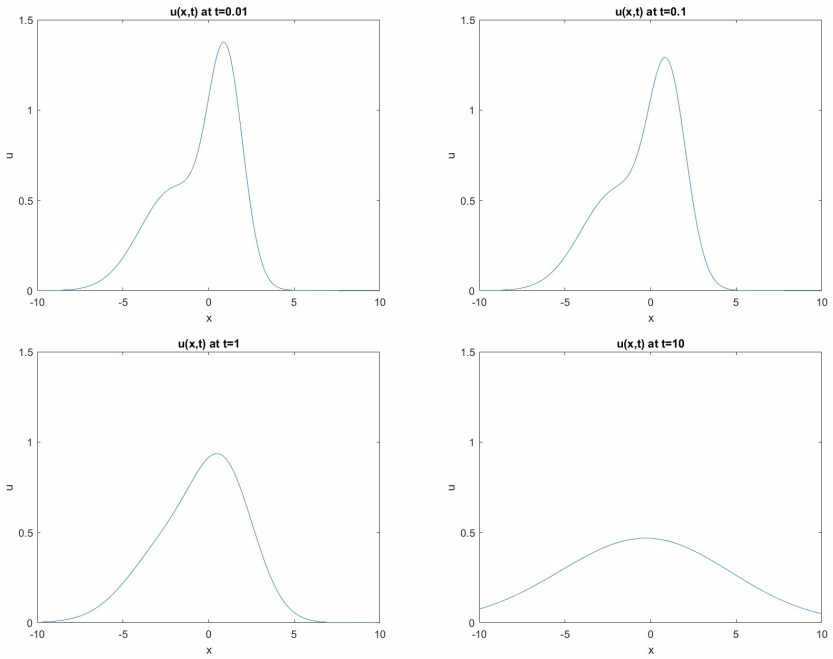

Figure 5: Solutions for Example 2 at different time values. Solutions are calculated numerically using the Heat Kernel and the trapezoid rule for $t = [0.01, 0.1, 1, 10]$.

The exact solution for this initial condition is,

$$u(x,t) = \frac{\frac{3\pi e^{\frac{-(x-1)^2}{2+4t}}}{\sqrt{\pi(\frac{1}{2}+t)}} + \frac{2\sqrt{2}\pi e^{\frac{-(x+2)^2}{8+4t}}}{\sqrt{\pi(2+t)}}}{2\pi}. \tag{21}$$

The largest error between our numerical method and this exact solution at any of the above time values is $1.99 \cdot 10^{-15}$.

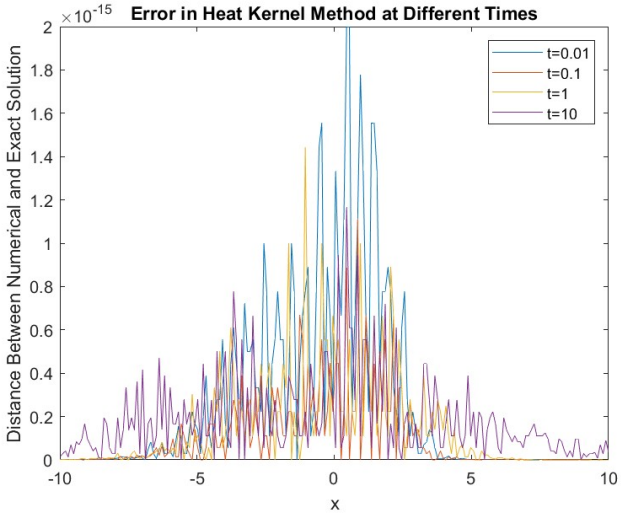

Figure 6: Error in numerical method for Example 2. Figure shows the absolute error as compared to the exact solution for the Heat Equation. These are compared at the same time values as in Figure 5.

3.4.3 Example 3

Next, we will consider the Heat Equation in two dimensions with a single Gaussian initial condition.

$$u_t - (u_{xx} + u_{yy}) = 0, \quad u(x, y, 0) = \frac{1}{\sqrt{2\pi}} e^{\frac{-(x+y)^2}{2}}. \tag{22}$$

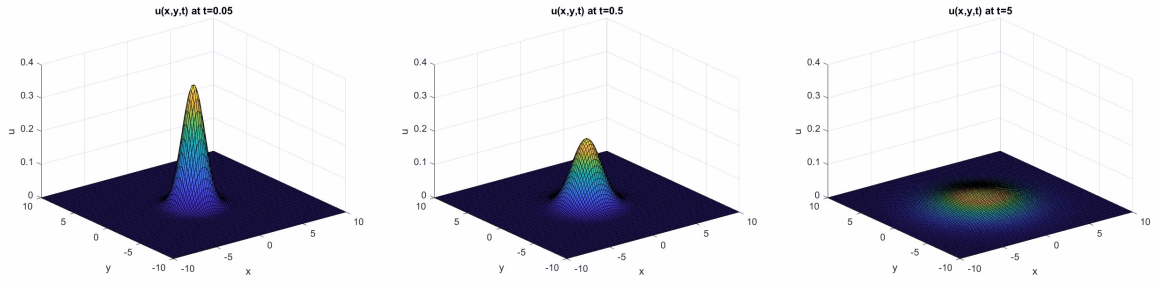

Figure 7: Solutions for Example 3 at different time values. Solutions are calculated numerically using the Heat Kernel and the trapezoid rule, now in two dimensions, for $t = [0.05, 0.5, 5]$.

The exact same method is used in two dimensions. Integration from (17) is now done in the x and y dimensions instead of just x.

3.4.4 Example 4

Finally, we will consider the Heat Equation in two dimensions with a a superposition of Gaussian functions.

$$u_t - (u_{xx} + u_{yy}) = 0, \quad u(x, y, 0) = \frac{3}{\sqrt{2\pi}} e^{\frac{-((x-1)^2+(y-1)^2)}{2}} + \frac{2}{\sqrt{2\pi}} e^{\frac{-((x+2)^2+(y+2)^2)}{2}}. \tag{23}$$

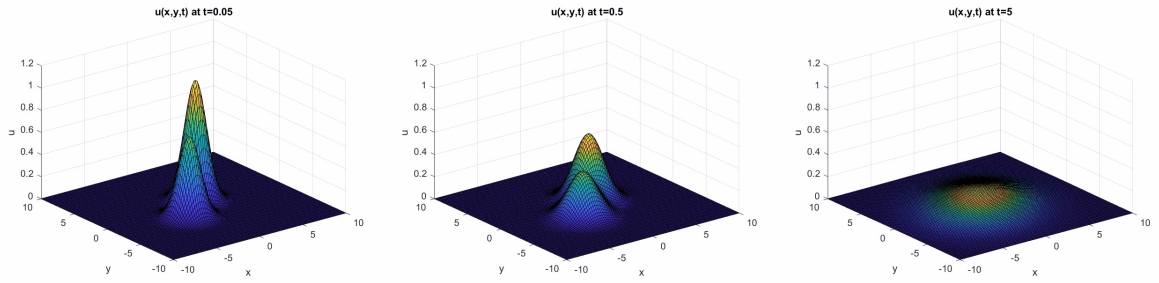

Figure 8: Solutions for Example 4 at different time values. Solutions are calculated numerically using the Heat Kernel and the trapezoid rule for $t = [0.05, 0.5, 5]$.

4 Green's Function for the PWE

Our next task will be to extend this method to the homogeneous PWE and compare against the well-known exact solutions for this partial differential equation. From this experience, we will then convert the nonhomogeneous PWE to an integral equation. Our goal then is to used discretized methods to this integral equation to obtain accurate, approximate solutions, and eventually study convergence results of our numerical methods.

4.1 Derivation of Green's Function for the PWE

The Heat Equation and the PWE are connected via the change of variables $t = \frac{iz}{2k}$. After completing this change of variables, we have the very similar IVP,

$$\Delta u + 2ik\frac{\partial u}{\partial z} = 0, \quad u(x, y, 0) = \phi(x, y). \tag{24}$$

Just as in section 3.1 when we derived the Heat Kernel, we will let $\mathscr{F}[u] = U(\omega_x, \omega_y, z)$ be the Fourier transform of u. Then, taking the Fourier transform of the PWE, we have,

$$\begin{aligned}
\frac{\partial}{\partial z}U - \frac{\omega_x^2 + \omega_y^2}{2ik}U &= 0 \\
\frac{\partial}{\partial z}U &= \frac{\omega_x^2 + \omega_y^2}{2ik}U \\
U &= U(\omega_x, \omega y, 0)e^{\frac{\omega_x^2+\omega_y^2}{2ik}z} \\
U &= \mathscr{F}[\phi(x, y)]e^{\frac{\omega_x^2+\omega_y^2}{2ik}z}.
\end{aligned}$$

Then we have our solution u,

$$u(x, y, z) = \mathscr{F}^{-1}[\mathscr{F}[\phi(x, y)]e^{\frac{\omega_x^2+\omega_y^2}{2ik}z}].$$

Again exploiting the convolution of Fourier transforms, we have,

$$u(x, y, z) = \phi(x, y) * \mathscr{F}^{-1}[e^{\frac{\omega_x^2+\omega_y^2}{2ik}z}].$$

Then our final Green's function G is equal to,

$$\begin{aligned}
G(x, y, z) &= \mathscr{F}^{-1}[e^{\frac{\omega_x^2+\omega_y^2}{2ik}z}] \\
&= \frac{iz}{2k}e^{\frac{iz}{2k}(x^2+y^2)}.
\end{aligned}$$

5 PWE Exact Solutions

Several exact solutions of the (homogeneous) PWE are known and given in Chapter 4 of Andrews and Phillips, [AP05], which we now quote. These exact solutions will serve as benchmarks for the numerical methods we will develop.

1. (Gaussian Beam Solution) Start with the Gaussian initial condition

$$u(x,y,0) = \exp(-\frac{x^2+y^2}{W_0^2})\exp(-i\frac{k(x^2+y^2)}{2F_0}).$$

Then the following is the exact solution to PWE:

$$u(x,y,z) = \frac{1}{\sqrt{\Theta_0(z)^2+\Lambda_0(z)^2}}\exp(-\frac{x^2+y^2}{W(z)^2})\exp(-i(\phi(z)+\frac{k(x^2+y^2)}{2F(z)})),$$

where

$$\Theta_0(z) = 1-\frac{z}{F_0}, \quad \Lambda_0(z) = \frac{2z}{kW_0^2},$$

$$\phi(z) = \tan^{-1}\frac{\Lambda_0}{\Theta_0}, \quad W(z) = W_0\sqrt{\Theta_0^2+\Lambda_0^2}, \quad F(z) = \frac{F_0(\Theta_0^2+\Lambda_0^2)(\Theta_0-1)}{\Theta_0^2+\Lambda_0^2-\Theta_0}$$

(see pages 92-93 of [AP05]). The functions ϕ, W and F are called the *longitudinal phase shift*, *spot size radius*, and *radius of curvature* at z, where z is the direction of propagation.

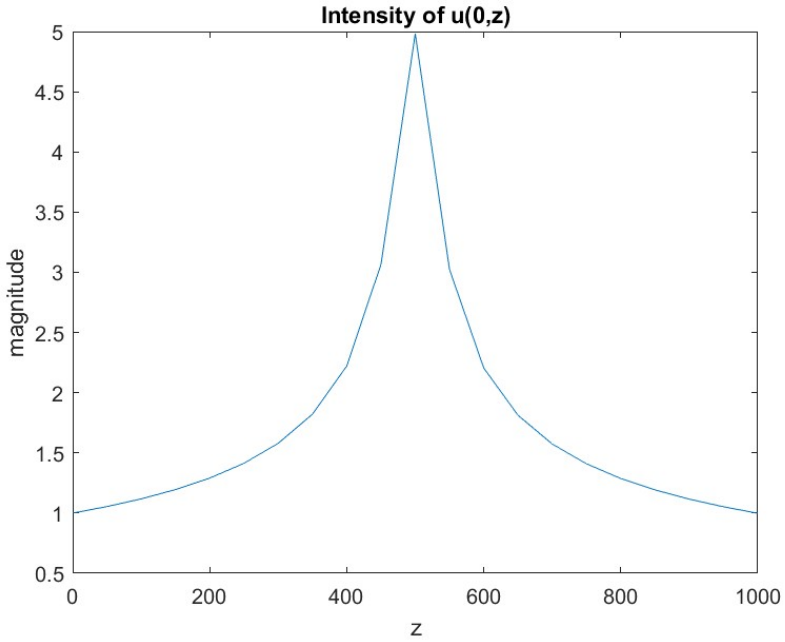

Figure 9: Intensity of 1D PWE at $x = 0$, where the intensity of u is defined as $|u|$. Note that the peak intensity occurs at $z = F_0$, where in this case $F_0 = 500$.

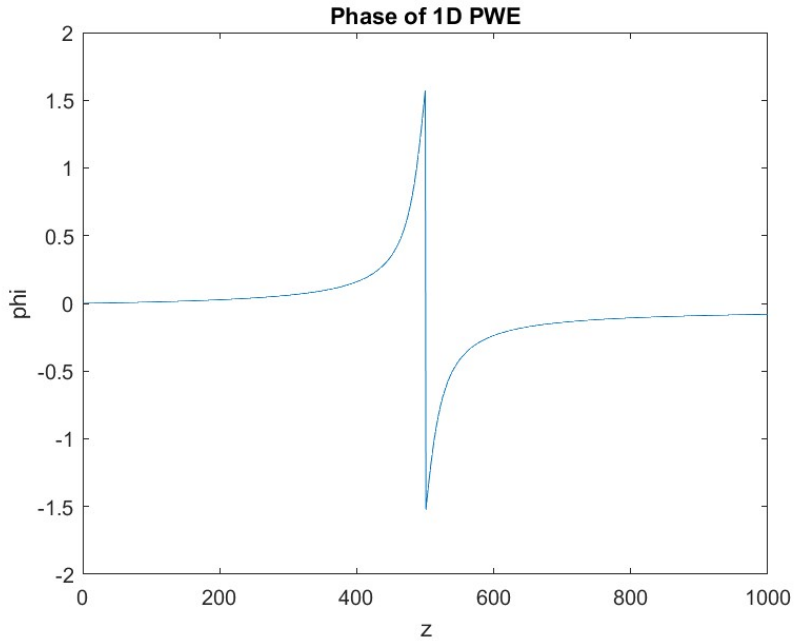

Figure 10: Phase of 1D PWE. Phase is defined as $\phi(z) + \frac{k(x^2+y^2)}{2F(z)}$, and is a function of x, y, and z.

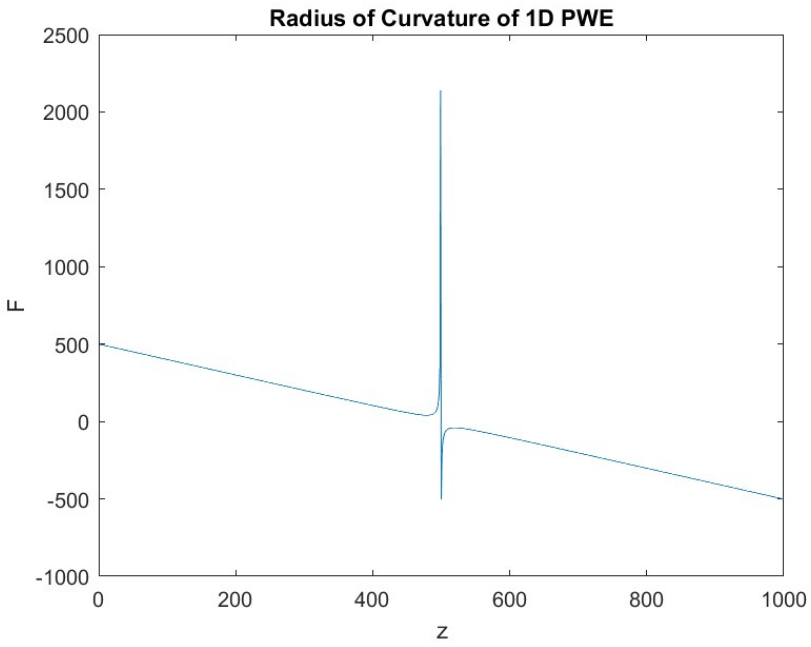

Figure 11: Radius of Curvature of 1D PWE. Radius of Curvature is defined above as F and is a function of z only.

2. (Hermite-Gaussian Beams) Let

$$u_{mn}(x,y,0) = H_m(\frac{\sqrt{2}x}{W_{x,0}})H_n(\frac{\sqrt{2}y}{W_{y,0}})e^{(-\frac{x^2}{W_{x,0}^2}-\frac{y^2}{W_{y,0}^2})},$$

where H_n is the n-th Hermite polynomial. Then

$$U_{mn}(x,y,z) = (\Theta - i\Lambda)(\frac{\Theta - i\Lambda}{\Theta + i\Lambda})^{\frac{m+n}{2}} H_m(\frac{\sqrt{2}x}{W}) H_n(\frac{\sqrt{2}y}{W}) e^{(\frac{ik}{2z}(\Theta + i\Lambda)(x^2+y^2))}$$

is the exact solution of the homogeneous PWE.

3. (Laguerre-Gaussian Beams) For $z = 0$, let

$$U_{mn}(r,\theta,0) = (\frac{\sqrt{2}r}{W_0})^m (-i)^m e^{im\theta} e^{-\frac{r^2}{W_0^2}} L_n^{(m)}(\frac{2r^2}{W_0^2}),$$

where W_0 is the radius of the TEM_{00} mode beam, $L_n^m(x)$ is the Laguerre polynomial, and n and m are the radial and angular mode numbers. Then

$$U_{mn}(r,\theta,z) = C_{mn}\frac{W_0}{W}(\frac{\sqrt{2r}}{W(z)})^m L_n^{(m)}(\frac{2r^2}{W^2(z)}) e^{-i(2n+m+1)\phi(z)} e^{(ikz + im\theta - \frac{r^2}{W^2} - i\frac{kr^2}{2F})}$$

is the exact solution of the homogeneous PWE.

6 PWE Numerics

This is the most important part of the project. The exact solutions discussed so far are secondary, and their main utility is to benchmark the Green's Function numerical method. We will now apply a new MATLAB code based on our same method from the Heat Equation, but now including this new change of variables to convert to the PWE.

Advantages of this numerical method are:

- It is highly accurate since it is essentially evaluation of an exact solution. The numerical error is only due to quadrature and this error can be controlled. There is no time-stepping (or rather z stepping) error.
- The solution can be evaluated at any time (or z) without having to very small time steps to avoid numerical instability (i.e. the Courant-Friedrichs-Lewy condition for convergence).
- The method does not require any matrix inversions, only convolutions. In principle, the method could be accelerated by the Fast Multipole Method (although we do not do that).

6.1 Numerical Integration

Instead of the Trapezoid Rule, we are now using the built-in MATLAB numerical integration function "integral." These integrals are being calculated from -20 to 20 for the dummy integration variable, with an absolute error tolerance of $1 \cdot 10^{-10}$. This built-in function uses a global adaptive quadrature method, and helps to solve the issue of integral values exceeding storage value limits when z is extremely close to 0.

6.1.1 Example 5

We are considering the problem,

$$\Delta_\perp u + 2ik\frac{\partial u}{\partial z} = 0, \quad u(x,y,0) = e^{\frac{-x^2}{W_0^2}} \cdot e^{\frac{-ikx^2}{2F_0}}. \tag{25}$$

We will start by analyzing this problem in just one dimension, and will analyze the PWE over $-0.1 \leq x \leq 0.1$ and $0 \leq z \leq 1000$. Let $W_0 = 0.05$ meters, $F_0 = 500$ meters, and $k = \frac{2\pi}{\lambda}$, where $\lambda = 633$ nanometers. These constants are realistic approximations for real-world conditions. As seen in Figure 9, the solution to the PWE appears very different from the solution to the Heat Equation. Rather than the Gaussian initial condition solely decaying over time, we now see wave-like behavior where the Gaussian beam contracts and then expands again.

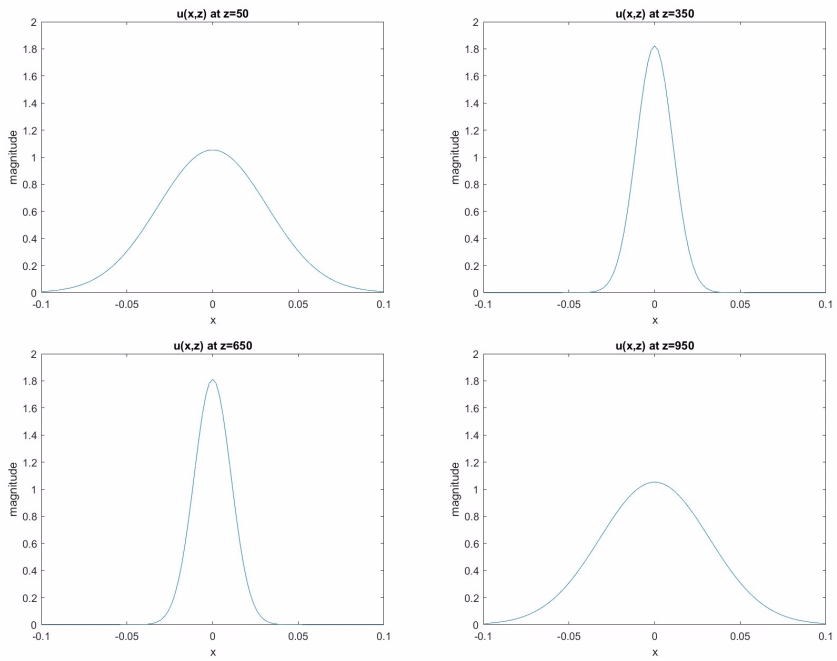

Figure 12: Solutions for Example 5 at different z-values. Solutions are calculated numerically using the PWE Kernel and the built-in MATLAB function integral for $z = [50, 350, 650, 950]$. Note how $|u|$ now demonstrates wave-like behavior in z, as opposed to simply decaying in time.

Displayed here is the intensity of the beam, $|u(x, z)|$. This now takes into account both the real and imaginary parts of the solution u, as opposed to the solely real parts found in the Heat Equation.

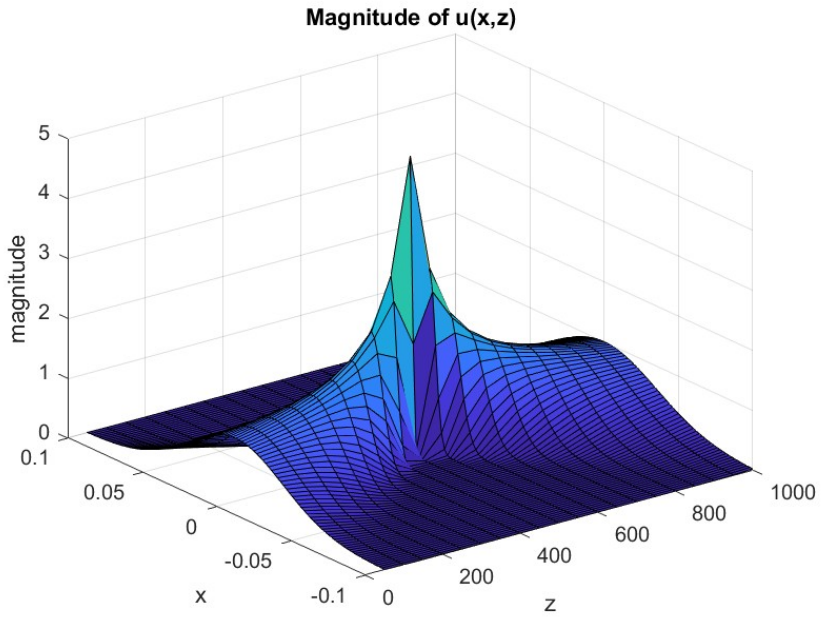

Figure 13: Surface Plot of Example 5 solution. Solutions are calculated numerically using the PWE Kernel and the built-in MATLAB function integral, but now across the full range of z-values.

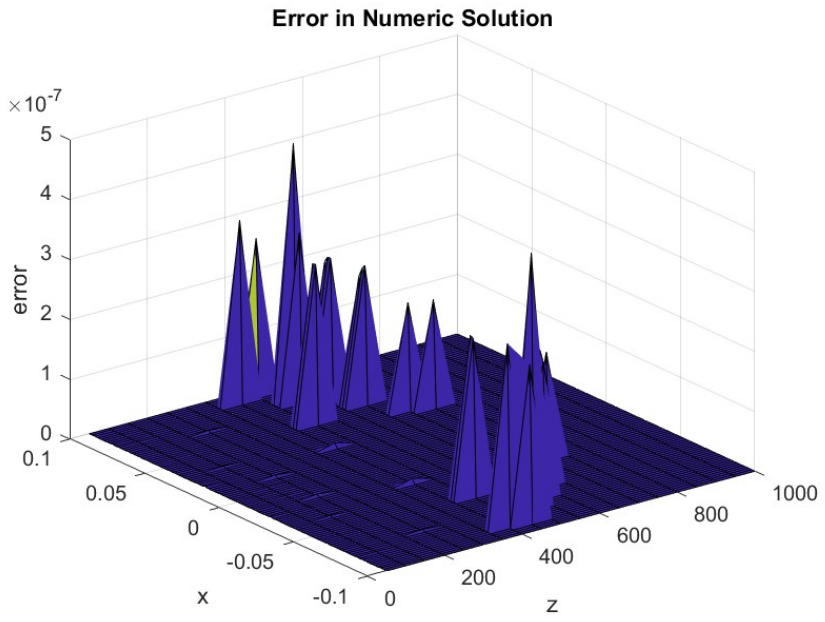

Figure 14: Error in Example 5 numerical solution as compared to the PWE exact solution found in Andrews and Phillips.

6.2 Non-dimensionalization

At this point, we want to non-dimensionalize our problem. This means that we will scale x, y, and z by set factors. Let

$$x = \frac{\bar{x}}{W_0}, \qquad y = \frac{\bar{y}}{W_0}, \qquad z = \frac{\bar{z}}{F_0}.$$

The goal of scaling x and y by W_0 and z by F_0 is twofold. First, we want to make the problem easier to interpret. By non-dimensionalizing in z, for example, we know that the beam will have its highest intensity at $z = 1$, as opposed to at $z = F_0$. Secondly, this will also simplify our numerics by allowing us to work with numbers closer to unity. The PWE now takes the form:

$$\Delta_\perp u + \alpha \frac{\partial u}{\partial z} + \beta(x, y, z)u = 0,$$

where

$$\alpha = \frac{2ikW_0^2}{F_0}, \quad \beta = W_0^2 k^2 (\frac{n(r)^2}{n_0^2} - 1).$$

7 Nonhomogeneous Case

We will now approach the case in which the index of refraction is variable.

7.1 Generation of the n-field

Professor Nick Moore has helped to produce a code which creates a realistic refractive-index field in two dimensions. It begins by creating a grid of wavenumber k values, and then applies Kolmogorov's theory of turbulence, [Bat67]. According to this theory, the power spectrum exhibits a -5/3 power law, and so the wavenumber grid is raised to the -5/3 power and then applied to another grid of random phase angles. We then transform back to physical space via the inverse Fourier transform to get the perturbation to the n-field.

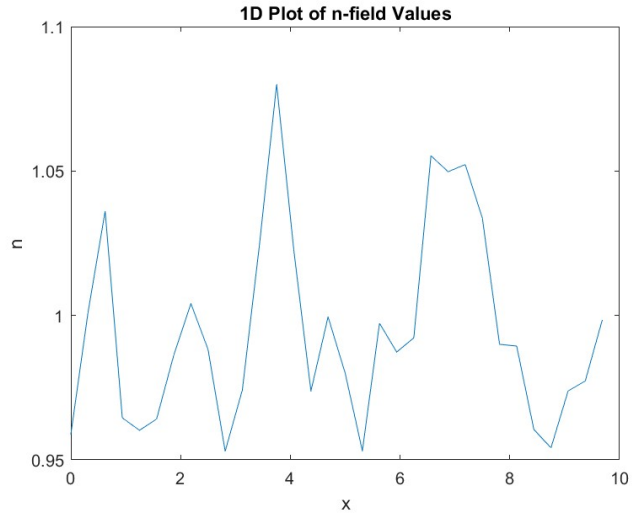

Figure 15: 1D Plot of an example refractive index field. Note that most values are between 0.95 and 1.05, and that values closer to 1 are more similar to the reference refractive index which represents propagation through a vacuum.

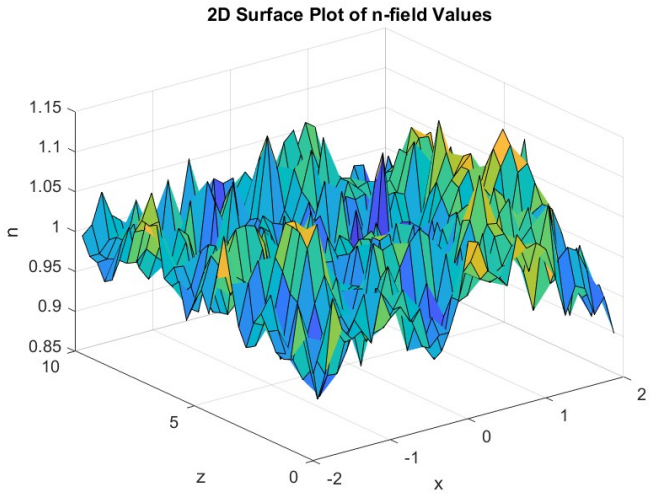

Figure 16: 2D Surface Plot of an example refractive index field. This shows the variation in both x and z.

These fields can be adjusted using a single ϵ value to scale the perturbation field, and can be generated over any x and z grids to any desired precision. This sample code is in Section 9.3 below.

7.2 Perturbation

Our new, full PWE problem is now,

$$\Delta_\perp u + 2ik\frac{\partial u}{\partial z} + k^2(\frac{n^2}{n_0^2} - 1)u = 0, \quad u(x, y, 0) = \phi(x, y). \tag{26}$$

We will define a new function f and constant ϵ, where

$$\epsilon f(x, y, z) = k^2(\frac{n^2}{n_0^2} - 1).$$

Here, ϵ is a small parameter and f is a function of x, y, and z. This gives us the nonhomogeneous PWE

$$\Delta_\perp u + 2ik\frac{\partial u}{\partial z} + \epsilon f(x,y,z)u = 0, \quad u(x,y,0) = \phi(x,y). \tag{27}$$

The parameter ϵ in (27) allows us to view the $\epsilon f(x,y,z)u$ as a small perturbation of the remaining terms in (27). With that interpretation, we denote the solution u in (27) as $u(x,y,z,\epsilon)$ and expand it in ϵ about $\epsilon = 0$:

$$u(x,y,z,\epsilon) = U_0(x,y,z) + \epsilon U_1(x,y,z) + \epsilon^2 U_2(x,y,z) + \cdots. \tag{28}$$

Here, following the usual formulas in a Taylor series of a function of ϵ, $U_0(x,y,z) = u(x,y,z,0)$, and as we will see shortly, U_0 is nothing but the solution of the homogeneous problem. The higher order terms are related to the usual derivatives of u with respect to ϵ, but since we don't u, those derivative formulas are useless. Instead, we follow a different strategy to find U_i, $i > 0$. Before proceeding, we demand that $U_0(x,y,0) = \phi(x,y)$, the initial condition of the full problem in (27), and $U_i(x,y,0) = 0$ for $i = 1, 2....$

Substituting (28) into (27) and collecting ϵ terms:

$$0 = \Delta_\perp(U_0 + \epsilon U_1 + \epsilon^2 U_2 + \cdots) + 2ik\frac{\partial}{\partial z}(U_0 + \epsilon U_1 + \epsilon^2 U_2 + \cdots) + \epsilon f(U_0 + \epsilon U_1 + \epsilon^2 U_2 + \cdots), \tag{29}$$

and then rearranging,

$$0 = (\Delta_\perp U_0 + 2ik\frac{\partial U_0}{\partial z}) + \epsilon(\Delta_\perp U_1 + 2ik\frac{\partial U_1}{\partial z} + fU_0) + \epsilon^2(\cdots). \tag{30}$$

This brings us back to the initial-value problem of the homogeneous PWE for U_0,

$$\Delta_\perp U_0 + 2ik\frac{\partial U_0}{\partial z} = 0,$$

as well as the first order correction

$$\Delta_\perp U_1 + 2ik\frac{\partial U_1}{\partial z} = -fU_0, \quad U_1(x,y,0) = 0.$$

The solution to U_1

$$U_1(x,y,z) = -\int_0^z G * fU_0\, ds = -\int_0^z \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(x-\xi, y-\eta, z-s) f(\xi,\eta,s) U_0(\xi,\eta,s)\, d\xi d\eta ds, \tag{31}$$

where $G(x,y,z)$ is the Green's function for the homogeneous problem, and $G * \phi$ is the solution to the homogeneous problem.

7.3 Numerical Approach

Since we now have a realistic f from the generated n-field, all that remains is to perform this convolution and integration. The convolution will be done over x and y as before. However, this time we will not have an exact solution for $G * fu_0$ to use as a benchmark. Our goal will be to calculate:

$$U_1(x,y,z) = -\int_0^z \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(x-\xi, y-\eta, z-s) f(\xi,\eta,s) U_0(\xi,\eta,s)\, d\xi\, d\eta\, ds. \tag{32}$$

To begin, we will discretize the outer z-integral by using a simple right hand rule. We define $z_i = i\Delta z, i = 1, \cdots, N$, where $\Delta z = z/N$. This gives

$$U_1(x,y,z) \approx -\sum_{i=1}^{N} \Delta z \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(x-\xi, y-\eta, z_N - z_i) f(\xi,\eta,z_i) u_0(\xi,\eta,z_i)\, d\xi\, d\eta, \tag{33}$$

where z_N will be our intended z-point at which to find a solution.

7.4 Example 6

We are considering the problem,

$$\Delta_\perp u + 2ik\frac{\partial u}{\partial z} + k^2(\frac{n^2}{n_0^2} - 1)u = 0, \quad u(x, y, 0) = e^{-x^2} \cdot e^{\frac{-ikW_0^2x^2}{2F_0}}. \tag{34}$$

We are now analyzing the full, nonhomogeneous PWE in two dimensions. We will analyze the PWE over $-4 \leq x \leq 4$, $-4 \leq y \leq 4$ and $0 \leq z \leq 2.4$. These ranges, now non-dimensionalized, have been chosen to demonstrate the behavior of the solution, while remaining computationally feasible. We are using a grid of 256 points in x and y, with a grid of 48 points in z, and are using the trapezoid quadrature rule once again in order to calculate the integral equation from (33).

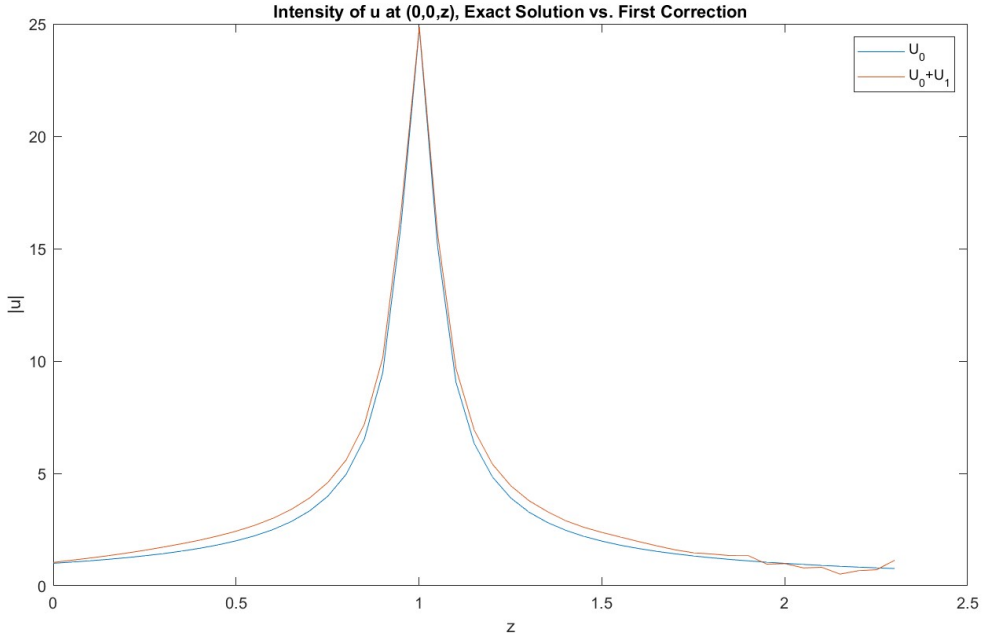

Figure 17: The homogeneous exact solution U_0 compared with the first-order correction $U_0 + U_1$ across the range of z-values for $x = 0$, $y = 0$.

We observe that the intensity of the beam is affected only slightly by the nonhomogeneous term, and that the intensity is sometimes greater and sometimes less than the homogeneous solution U_0 now with the addition of the first order correction.

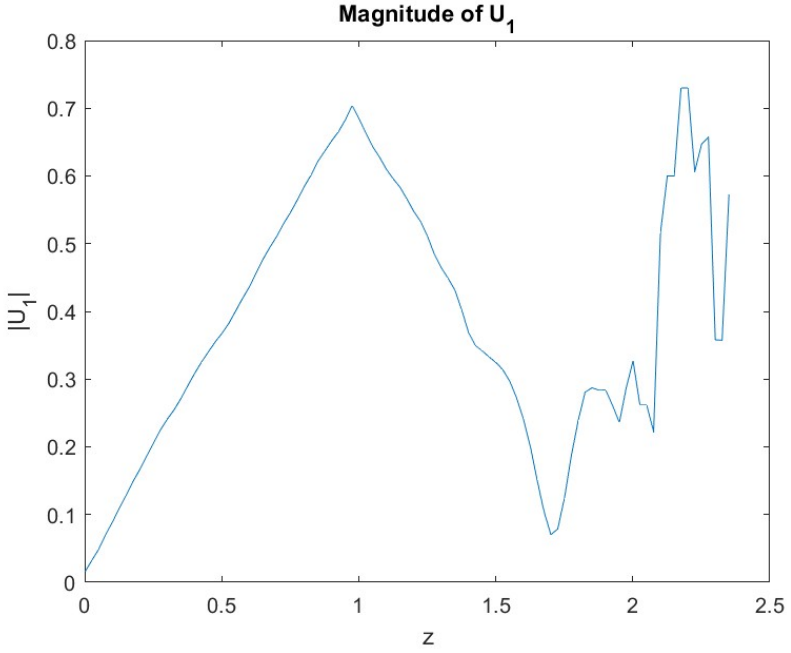

Figure 18: The magnitude of U_1 across the range of z-values for $x = 0$, $y = 0$. Note that the maximum magnitude for U_1 is less than one.

8 Conclusion

We have shown that we have a working Green's Function method for finding solutions to the Paraxial Wave Equation. We have also created working programs to compute solutions to the homogeneous problem as well as the first order correction for the nonhomogeneous problem.

What remains is to analyze the behavior of these solutions as we vary different parameters. These could include beam parameters, such as the initial spot size W_0 and the initial radius of curvature F_0. They could also include perturbation parameters, such as the scale of the refractive-index field or the size of the perturbation parameter ϵ. Our same method can also be iterated in order to find the next correction term U_2. Much of this further study will need to be carried out experimentally. Physical data from real lasers propagating through turbulent media could be compared to our simulated results in order to find appropriate values for the perturbation parameters.

8.1 Future Research

Other valuable future research questions include:

1. Is our version of the PWE the right reduced model? Are all of the physical assumptions in Section 2 appropriate?
2. Does this method outperform possible alternatives, such as the spectral method?
3. What role does the strength of turbulence play in the numerical calculations?
4. What role does the phase play in this problem?
5. Can one introduce "noise" in the initial condition to mitigate the impact of turbulence in the medium?
6. How well will this approach work for different beams (Bessel, Laguerre, Hermite)?

8.2 Acknowledgment

Thank you very much to Reza Malek-Madani and Nick Moore for their guidance, mentorship, and good humor throughout this project. Thank you as well to Professor Anastasios Liakos and Professor Ana Maria Soane for their help in reading drafts and providing feedback.

9 Sample MATLAB Codes

9.1 Example 1 Code

```
% Heat Kernel method of finding solution
eps = 1; pts = 200;
% Epsilon replaces k to avoid confusion and set to 1, 200 x-values
K = @(x,t) (1/sqrt(4*pi*eps*t))*exp((-x^2)/(4*eps*t)); % Heat Kernel
init = @(x) (1/sqrt(2*pi))*exp(-(x^2)/2); % Gaussian initial condition
x = linspace(-10,10,pts);
times = [0.01, 0.1, 1, 10];
u = [0];

for j = 1:4
    t = times(j);
    for i = 1:pts
        u(j,i) = mytrap2(@(y) init(y)*K(x(i)-y,t),-20,20,1000);
        % Performs the convolution and integrates using 1000 trapezoids
    end
end

plot(x,u(1,:))
ylim([0,0.5]); xlabel('x'); ylabel('u'); title('u(x,t) at t=0.01');
figure
plot(x,u(2,:))
ylim([0,0.5]); xlabel('x'); ylabel('u'); title('u(x,t) at t=0.1');
figure
plot(x,u(3,:))
ylim([0,0.5]); xlabel('x'); ylabel('u'); title('u(x,t) at t=1');
figure
plot(x,u(4,:))
ylim([0,0.5]); xlabel('x'); ylabel('u'); title('u(x,t) at t=10');


% Exact Solution
u_exact = zeros(4,pts);
exactsolution = @(x,t) (1/(2*sqrt(pi*(t+0.5))))*exp(-(x.^2)/(2+4*t));
for j = 1:4
    t = times(j);
    u_exact(j,:) = exactsolution(x,t);
end

diff_u = abs(u-u_exact);
figure
for j = 1:4
    plot(x,diff_u(j,:))
    hold on
end
title('Error in Heat Kernel Method at Different Times')
legend('t=0.01','t=0.1','t=1','t=10')
xlabel('x'); ylabel('Distance Between Numerical and Exact Solution')
```

9.2 Example 5 Code

```
 % PWE with Green's Function method of finding solution
k = (2*pi)/(633*(10^-9));
x_pts = 101;
z_pts = 21;
x = linspace(-0.1,0.1,x_pts);
z = linspace(0,1000,z_pts);
W = 0.05; F = 500;
q = zeros(x_pts,z_pts); q_exact = zeros(x_pts,z_pts);

init = @(x) exp(-(x.^2)./(W^2)).*exp(-(1i*k*x.^2)./(2*F));
kernel = @(x,z) (sqrt(k)./sqrt(2*pi*1i*z)).*exp((1i*k*x.^2)./(2*z));

for z_int = 1:z_pts
    for x_int = 1:x_pts
        q(x_int,z_int) = integral(@(xi) init(xi).*kernel(x(x_int)-xi,z(z_int)),-20,20);
    end
end

q_real = real(q);
q_imag = imag(q);
q_mag = sqrt((q_real).^2 + (q_imag).^2);
figure
plot(q_mag)
title('magnitude of u, numeric solution');

% Computing exact solution from Mathematica
exactsolution = @(x,z) exp(-(x.^2)./(2*((1i*z/k) + (F*W^2)./(2*F + 1i*k*W^2)))).(sqrt((1i*k/F)+(2/W^
for z_int = 1:z_pts
    q_exact(:,z_int) = exactsolution(x,z(z_int));
end

q_real_exact = real(q_exact);
q_imag_exact = imag(q_exact);
q_mag_exact = sqrt((q_real_exact).^2 + (q_imag_exact).^2);

error = abs(q_mag-q_mag_exact);
surf(error)

figure
plot(z,q_mag_exact(51,:))
xlabel('z'); ylabel('magnitude'); title('Intensity of u(0,z)')

figure
plot(x,q_mag(:,2))
ylim([0,2]); xlabel('x'); ylabel('magnitude'); title('u(x,z) at z=50');
figure
plot(x,q_mag(:,8))
ylim([0,2]); xlabel('x'); ylabel('magnitude'); title('u(x,z) at z=350');
figure
plot(x,q_mag(:,14))
ylim([0,2]); xlabel('x'); ylabel('magnitude'); title('u(x,z) at z=650');
figure
plot(x,q_mag(:,20))
ylim([0,2]); xlabel('x'); ylabel('magnitude'); title('u(x,z) at z=950');
```

```
figure
surf(z,x,q_mag);
xlabel('z'); ylabel('x'); zlabel('magnitude'); title('Magnitude of u(x,z)');
figure
surf(z,x,error);
xlabel('z'); ylabel('x'); zlabel('error'); title('Error in Numeric Solution');
```

9.3 Example 6 Code

```
    %% GOAL: Illustrate how to do the z-integration
%% to model laser propogation through an inhomogeneous medium.
clear all
global W0 k F0 weight n i deltaxi deltaeta xvalues zvalues a c Nz

% Parameters:
n=256;W0=0.05;F0=500;k=2*pi/(633*10^-9);deltaxi=8/(n-1);deltaeta=deltaxi;
a=-4;c=a;
xvalues = linspace(a,-a,n);
zvalues = 0.001:0.025:2.4;
Nz = max(size(zvalues));
zfin = max(zvalues);      % The final value of z at which to evaluate the solution.

% To compute:
% u1(z)         The first-order correction evaluated at a single z value.
% The grid of z values for the z integration.
dz = (2.351-0.001)/Nz;
yvalues = xvalues;

% Step 1. Obtain leading-order solution u_0(x,z) by solving IVP.
% To save time, using A&P exact solution
u_0 = zeros(n);
test_1=[];
for i = 1:Nz
    for j = 1:n
        % u_0 is a "3D" matrix, format (x,y,z)
        u_0(:,j,i) = u0_cond(xvalues,yvalues(j),zvalues(i));
    end
    test_1 = [test_1 u_0(n/2,n/2,i)];
end

% Step 2. Compute "inhomogeneous" term: k^2*(1-n^2/n0^2)*u_0
% fvals = nfield_gen .* u_0;
ngrid = nfield_gen;
fvals = zeros(n);
test_2=[];
for i = 1:Nz
    for j = 1:n
        % fvals is same format as u_0
        fvals(:,j,i) = u_0(:,j,i) .* ngrid(i,:)';
    end
    test_2 = [test_2 fvals(n/2,n/2,i)];
end

% Step 3. Perform z-integration to compute u_1(x,z)
u1 = 0;
corr_array=[];
% Currently ignoring final z-value, as G(x,y,0) is a singularity
```

```
for i = 1:Nz-1
    tic
    zval = zvalues(i)
    % Takes slice of fvals at z-index, evaluates at zfin-zval
    corr_val = green_solve(fvals(:,:,i),zfin-zval);
    u1 = u1 + dz * corr_val;
    corr_array = [corr_array u1];
    toc
end

exact_x0y0 = u0_cond(0,0,zvalues(1:Nz-1));
figure
plot(zvalues(1:Nz-1), abs(exact_x0y0));
first_corr_x0y0 = exact_x0y0 + corr_array;
hold on
plot(zvalues(1:Nz-1), abs(first_corr_x0y0));
legend('U_0','U_0+U_1')
title('Intensity of u at (0,0,z), Exact Solution vs. First Correction');
xlabel('z')
ylabel('|u|')

figure
plot(zvalues(1:Nz-1), real(exact_x0y0));
hold on
plot(zvalues(1:Nz-1), real(first_corr_x0y0));
legend('U_0','U_0+U_1')
title('Real part of u at (0,0,z), Exact Solution vs. First Correction');
xlabel('z')
ylabel('u (real)')

function w=green_solve(fvals,zval)
global W0 k F0 weight n m deltaxi deltaeta a c xvalues zvalues Nz u_0

u_0 = fvals;
% for j = 1:n
%     u_0(:,j) = InitCondition(xvalues,xvalues(j));
% end

weight=ones(n,n);
weight(1,1)=0.25;weight(n,1)=0.25;
weight(1,n)=0.25;weight(n,n)=0.25;
for i=2:n-1
    weight(i,n)=0.5;
    weight(i,1)=0.5;
end
for j=2:n-1
    weight(n,j)=0.5;
    weight(1,j)=0.5;
end
w = solutionKernel(0,0,zval);
% term=[];
% for m = 1:Nz
%     tic
%     zvalues(m)
%     val_z = solutionKernel(0,0,zvalues(m));
%     toc
```

```
%     term = [term val_z];
% end
% plot(zvalues,abs(term))

function w=solutionKernel(x,y,z)
global W0 k F0 weight n m deltaxi deltaeta a c xvalues zvalues

sum=0;
% ngrid = nfield_gen';

for i=1:n
    for j=1:n
        sum = sum + weight(i,j) * integrand(x,y,z,a+(i-1)*deltaxi,c+(j-1)*deltaeta,i,j) * deltaxi * d
    end
end
w=sum * W0^2;

function w = u0_cond(xin,yin,zin)
global W0 k F0 weight n deltaxi deltaeta a c

    theta0=1-zin;lambda0=2*F0*zin/(k*W0^2);
    part1=theta0.^2+lambda0.^2;
    part2=sqrt(part1);

    phi=atan(lambda0./theta0);
    W=W0*part2;
    F=F0*part1.*(theta0-1)./(part1-theta0);
    w=1./part2.*exp(-W0^2*(xin.^2+yin.^2)./W.^2).*exp(1i*(-phi-k*W0^2*(xin.^2+yin.^2)./(2*F)));

end

function w = integrand(x,y,z,xi,eta,xi_index,eta_index)
global W0 k F0 weight n deltaxi deltaeta a c u_0

% u0_val = InitCondition(xi,eta);
u0_val = u_0(xi_index,eta_index);

green_val = GreenPWE(x-xi,y-eta,z);

w = - green_val .* u0_val;
end

function w=GreenPWE(x,y,z)
global W0 k F0 weight n deltaxi deltaeta a c

w=-1i*k./(2*pi*F0*z).*exp(1i*k*W0^2*(x.^2+y.^2)./(2*F0*z));
```

9.4 Refractive Index Field Setup Code

```
%% Goal: Create a realistic refractive-index field in 2D.
clear all; clc;

%-------------------------------------%
%% Set all parameters here.
% The number of grid points in x and z;
% npts needs to be even, ideally a power of 2.
npts_x = 32;
```

```
npts_z = 32;
% The distance to span in the z direction and x direction.
zend = 10.0;
xend = 2.0;
% The scale of the refractive-index perturbation field.
eps = 0.1;
%--------------------------------------%

% Set up the grid of wavenumber k values.
k1 = [0:npts_z/2-1, -npts_z/2:-1];
k2 = [0:npts_x/2-1, -npts_x/2:-1];
k1grid = k1' * ones(1,npts_x);
k2grid = ones(npts_z,1) * k2;
kabs_grid = sqrt(k1grid.^2 + k2grid.^2);

% Set up the power spectrum according to Kolmogorov.
pow_spec = (kabs_grid).^(-5/3);
pow_spec(1,1) = 0;
% Compute the transform nhat based on power spec.
theta = 2*pi*rand(size(pow_spec));
nhat = sqrt(pow_spec) .* exp(i*theta);
% Transform back to physical space to get the perturbation to n-field.
npert = npts_x*npts_z * ifft2(nhat);
% Calculate the refractive-index field.
nfield = ones(size(npert)) + eps/10 * real(npert);

% Set up the grid of x and z values.
dx = 2*xend/npts_x;
xvals = -xend:dx:(xend-0.5*dx);
dz = zend/npts_z;
zvals = 0:dz:(zend-0.5*dz);

% 2D Plot
figure();
[xgrid, zgrid] = meshgrid(xvals, zvals);
surf(xgrid, zgrid, nfield);
xlabel('x'); ylabel('z'); zlabel('n');
% 1D Plot as a check.
figure();
plot(zvals, nfield(:,1))
```

References

[AP05] Larry C. Andrews and Ronald L. Phillips. *Laser Beam Propagation Through Random Media*. The International Society for Optical Engineering, 2005.

[Bat67] G. K. Batchelor. *An Introduction to Fluid Dynamics*. Cambridge University Press, 1967.

[GS90] Leslie Greengard and John Strain. A fast algorithm for the evaluation of heat potentials. *Communications on Pure and Applied Mathematics*, 1990.

[Hab12] Richard Haberman. *Applied Partial Differential Equations With Fourier Series and Boundary Value Problems*. Pearson, 2012.

[JMMAZ22] Kyle Jung, Reza Malek-Madani, and Svetlana Avramov-Zamurovic. A fourier collocation method approach to solving the paraxial wave equation. 2022.

[MM98] Reza Malek-Madani. *Advanced Engineering Mathematics With Mathematica and Matlab*. Prentice Hall, 1998.

[Str78] John W. Strohbehn. *Laser Beam Propagation in the Atmosphere*. Springer, 1978.

[Str07] Walter A. Strauss. *Partial Differential Equations, An Introduction*. Wiley, 2007.

[TF48] Robert K. Tyson and Benjamin W. Frazier. *Field Guide to Adaptive Optics*. The International Society for Optical Engineering, 1948.