SIMULATION AND COMPARISON OF LASER BEAM PROPAGATION THROUGH A HIGH SPEED FLOW

Thesis

Submitted to

The School of Engineering of the

UNIVERSITY OF DAYTON

In Partial Fulfillment of the Requirements for

The Degree

Master of Science in Electro-Optics

By

Luke Alan Borntrager

UNIVERSITY OF DAYTON

Dayton, Ohio

August 2004

SIMULATION AND COMPARISON OF LASER BEAM PROPAGATION THROUGH A HIGH SPEED FLOW

APPROVAL PAGE

Dr. Andrew M. Sarangan, Ph.D.
Professor, Electro-Optics
Advisory Committee, Chairman

Dr. Bradley D. Duncan, Ph.D.
Professor, Electro-Optics
Committee Member

Dr. Stanley Rogers, Ph.D.
Major, USAF Reserve, AFLR/MLPS
Senior Research Scientist/Engineer,
EO CM Technology Branch, AFRL/SNJW
Professor & Undergraduate Student Advisor,
Electrical & Computer Engineering
Committee Member

Dr. Donald L. Moon, Ph.D.
Associate Dean
Graduate Engineering Programs &
Research School of Engineering

Dr. Joseph Saliba, Ph.D.
Dean
School of Engineering

ABSTRACT

SIMULATION AND COMPARISON OF LASER BEAM PROPAGATION THROUGH A HIGH SPEED FLOW

Name: Borntrager, Luke Alan

University of Dayton, 2004

Advisor: Dr. Andrew Sarangan

This research was aimed at studying the characteristics of laser beam propagation through a high speed turbulent medium and the aberrations caused in the beam. This was a joint DAGSI program, between the University of Dayton, the Ohio State University, and the Air Vehicles and Sensors Directorates of the Air Force Research Laboratories. The focus of this thesis is on the modeling and simulation of the optical beam. The model was created to validate the experimental data and to develop the capability for predicting aero-optic aberrations. The simulations were in agreement with the experimental results with a 90% correlation factor in nearly all cases.

Several methods of beam propagation were explored for use in the simulation, using commercial as well as custom codes. A series of validation tests were performed to verify the code's ability to accurately simulate conditions similar to those associated with the experiment. The conditions of the experiment were documented along with the recorded images of the high speed flow and corresponding wavefront information for

several trials. The images were then used as inputs to the computer code and simulated output phases were determined. These output phases were then correlated against the measured phases in the validation of the simulation. Simulations were then performed to explore the laser beam characteristics downstream.

ACKNOWLEDGEMENTS

I would like to thank a number of people who aided me in the quest in performing this research. First, I would like thank the University of Dayton's Electro-Optics department, in its entirety, for helping me in furthering my education. I would like to especially thank Dr. Andrew Sarangan, my advisor, who approached me with this research and has supported and guided me through out it as well as provided valuable advice on simulation and coding techniques. I thank Dr. Bradley Duncan, Dr. Scott Harris, and Jeff Widiker for their hard work in the designing of the Shack-Hartmann sensor to collect data for this research as well as providing support, suggestions, and advice in building the simulations. I thank Dr. Mo Samimy and Brian Thurow for their work in the designing of the high speed flow and their valuable inputs pertaining to the flow mathematics. I thank Mr. Joseph Koesters for allowing me to work this research into my working development plans here in the Sensors Directorate at the Air Force Research Laboratories. I thank Dr. Stanley Rogers for his acceptance to be on the reviewing committee along with Dr. Duncan and Dr. Sarangan.

I would like to thank my grandfathers, Lee Borntrager and Curt Rigney, whom have both passed on, and my father, Gary Borntrager, for teaching me the value of hard work and dedication to whatever it is I am working on and that "anything worth doing is worth doing right."

I especially want to thank my wife, Stephanie, for being there, in every thing, in every way, supporting me through the good times and the bad, who took my last name and has held it high with the respect that only a great wife could. I admire her for her strength, for her faith in me, for her faith in God, and for her delivery of our son, Wyatt, whose bright smiling face has helped me rearrange my priorities to those of a dad, the way only a child could.

And to whom I owe it all; my life, my family, my dedication, without whom none of what I have would exist; I thank you, our Father in Heaven.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Laser Applications

In today's high tech world, not a day goes by without thousands of lasers being put to use. From entertaining us with music and sights we hear and see from CD and DVD players, helping us checkout at the local grocery store, and cutting and welding materials, to improving our vision, performing delicate surgeries, and providing us with clear communications through fiber optical networks, lasers have become common commodities to our society. And while a relatively "younger" science, with just under fifty years of life, lasers are finding themselves put to uses that were once only in our dreams and on the silver screen.

We have all, at one time or another, been entertained by the fictions of "laser cannons", "tractor beams", or "ray guns" being used as powerful weapons. Today, however, driven by practicality and usefulness, these fictions are becoming realities. Research over the last half century has lead to the development of lasers becoming one of the most powerful tools ever used by scientists in the realm of military applications. Lasers are now used in a multitude of applications such as sensing, guidance, weaponry, and countermeasure technologies. Along with these uses, however, scientists must

overcome a wide variety of issues to maintain the integrity of the laser beams of these systems.

1.2 Laser Beam Propagation Through Random Media

In the applications mentioned above where the uses involve propagating lasers over a distance, in particular a distance through any type of random media, aero-optical distortion is generally the primary limiting factor in the effectiveness of the system. Random media is described as media whose basic properties are random functions of space and time, such as atmospheric conditions.[1] These random functions introduce fluctuations into propagating beams, causing chaos for the mentioned applications. For example, in U.S. Air Force applications, in typical tactical flight conditions, the efficiency of the laser energy directed to a potential target is estimated to drop by 30 to 70% due to the turbulent shear layer over the transmitting window of the aircraft.[2]

Though random media does play a large role in interfering with the laser beam propagation, it does not, however, eliminate the concept of using laser beams for a wide variety of military purposes. These uses are not only still being pursued; they are being achieved as a result of our better understanding of how aberrations are caused and to what degree they occur.

1.3 Understanding Beam Aberrations From Random Media

Though beam aberrations occur in one form or another in practically every use of lasers, it is the degree of the aberration which makes the difference. For example, in the

use of compact disc players, the reflected beam from the disc is read by a sensor. This sensor, while interpreting the valid reflected information written to the disk, also takes into account small deformations and changes in the beam. This is a built in tolerance of aberration for the system. Likewise, if one knows the amount of aberration to the beam, it can be analyzed whether or not this amount of aberration can be tolerated for the particular application.

One particularly technology that helped overcome the amount of aberration is the use of corrective measures to overcome these counterproductive influences. The approach of using two deformable mirrors for correcting scintillation effects in laser beam projection through the turbulent atmosphere has been demonstrated. The concept uses a deformable mirror and a Fourier-transforming mirror to adjust the amplitude of the wave front. A second deformable mirror is used to correct the phase of the wave front before it leaves the aperture. Simulations of the beam propagation through a single turbulent layer sufficiently distant from the sensor and laser beam transmission aperture, to cause scintillation, show that the approach improves the on axis field amplitude significantly.[3] However, in order to use this type of system, near exact information is needed in real-time. This involves a costly system composed of a transmitter and a receiver on opposite sides of the medium causing the aberrations as well as sending the information to the deformable mirrors.

While this system proves to be limited in use because of cost and developing technologies, it doesn't spell doom for the aforementioned systems and concepts. Again, it goes back to the application and their tolerance for aberration to determine whether or not the use is valid. For example, in the idea of a system designed to direct a certain

magnitude of energy from a laser beam to a target area, if the approximate aberration of the laser beam being used in the particular situation is known, the loss of energy from the aberration can be calculated and the originating beam power can be increased to allow for the loss. Another example is the use of a modulated beam. If the amount of aberration is known, the phase differences can be calculated and the frequency at which the beam can be modulated without corruption can be determined. The system would either have to be modified to accept the condition or operate around the determined frequencies. In either of these two cases, the key is to know the amount of aberration over the medium being traversed.

1.4. Simulation of Aero-Optic Aberration

To aid in the study the study of these types of aberrations, a proposal was submitted to and funded by the Dayton Area Graduate Studies Institute[4] (DAGSI) titled "Measurement and Modeling of Aero-Optical Aberrations in Coherent Laser Radiation"[5] as part of a joint effort between the University of Dayton[6] (UD), the Ohio State University[7] (OSU), the Air Vehicles Directorate[8] of the Air Force Research Laboratories (AFRL/VA), and the Sensors Directorate[9] of the Air Force Research Laboratories (AFRL/SN). While the focus of the overall research pertains to the prediction and control of aero-optic aberrations, the research was broken down into three main areas. The first area was to measure the wavefronts of a laser beam passing through a turbulent stream by using a high speed charge coupled device (CCD) camera along with a Shack Hartmann wavefront sensor. The second area was to use modeling and simulation to provide means of predicting the amount of aberration introduced into the laser beam as it

passes through the turbulence. The third area was the formation of some method of controlling the amount of aberrations that the beam incurs.

The focus of this research is the second area of the above mentioned effort, to accurately simulate the propagation of an optical beam through a turbulent medium. A survey of existing beam propagation software was conducted and computer code was written in conjunction with commercial software to develop a beam propagation simulation tool capable of analyzing laser beam properties before, during, and after the propagation through a turbulent layer. Condensation maps of turbulence were recorded in part one of the above experiment and corresponding wavefront measurements data were taken.[10] The work detailed in this thesis involves taking these maps and converting them to approximate maps of indices of refraction and using them in the beam propagation model. These resulting wavefronts are then compared to the measured wavefronts.

The subject matter of this research is laid out in the following format. First, relevant information from the joint OSU/UD/AFRL study will be presented. This will entail a brief discussion of the techniques used to produce the turbulent jet stream as well as how the stream was analyzed and how data was obtained in the recording experiments. Next, a thorough section on the theory of beam propagation is presented, followed by a detailed explanation of how the information from the previous research efforts is used. A study of the different beam propagation software available at the time of this research is given, as well as the final choice the software. Finally, the simulation code is verified by computing the correlation between the measured data and the modeled output. The model parameters and their effects on the simulation results are investigated. Finally, it is

the ultimate goal of this thesis to develop a better the understanding of the aspects of random media that create the aberrations, and provide a model capable of replicating the experimental measurements.

CHAPTER 2

HIGH SPEED FLOW MEASUREMENTS

2.1 Overview

While the main focus of this thesis is the simulation of the beam propagation through a high speed flow, it does not deal with creating a flow model for the fluid. However, the flow must be represented in the simulation. For this purpose, information recorded by other members of this joint DAGSI project between OSU, UD and AFRL, will be utilized. Because of this, a chapter will be dedicated here to the discussion of the turbulent, high speed flow used in the experiment and the methodology for producing the flow.

Additional information will be provided in this chapter to discuss the efforts involved with recording the passage of the beam through the flow and documenting the output to be used in association with this research. While this chapter will include some minor mathematical explanations, it is not intended to be a full exploration into the details of recording the wavefronts. Rather, this chapter is meant to give a brief insight into how the images and wavefronts were obtained and to what extent the model must be prepared in order to simulate the recording experiments. For a detailed description of wavefront recording of this project, the reader is referred to "Design & Application of a High-Speed Shack-Hartmann Wavefront Sensor".[10]

2.2 Flow Production

To begin, a brief discussion of the formation of the jet stream needs to be accomplished. Insights into the type of flow that the beam is propagating through will help us understand how the model will be utilized in this experiment.

The recording experiment took place at the Ohio State University's Gas Dynamics and Turbulence Laboratory (GDTL).[11] To form the high speed flow, air is pumped into a stagnation chamber from two, four-stage compressors. The air is filtered, dried, and stored in two cylindrical tanks with a total capacity of 42.5 m^3 at 16.5 MPa or 1600 ft^3 at 2500 psi.[12] The stagnation chamber contains a perforated plate and formation where water contained in the warm moist ambient air condenses upon entrainment into the jet, where it mixes with the cold, dry air of the jet core. While adaptable to more than one jet nozzle, the nozzle used in this experiment, seen in figure 2.1, has a design to produce a Mach number of 1.3.

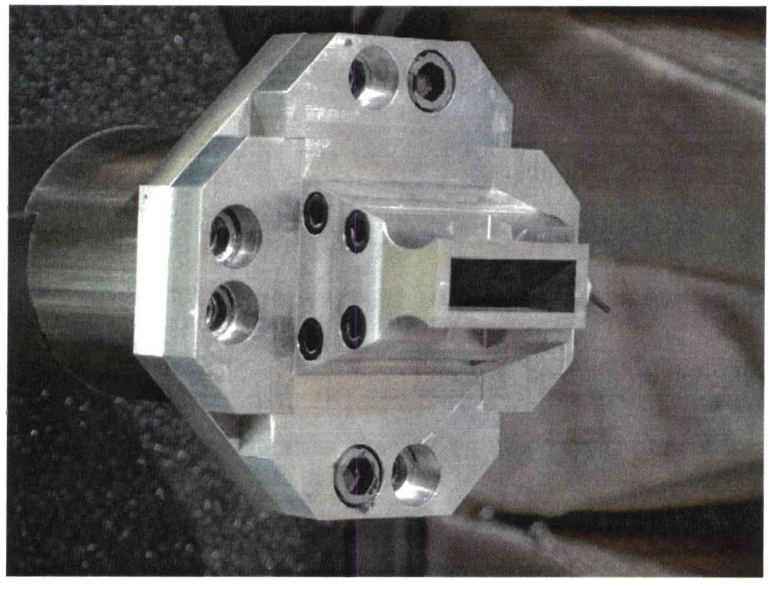

Figure 2.1 Mach 1.3 Jet Nozzle[13]

The nozzle itself has an interior exit diameter of 25.4 mm by 63.5 mm with a lip of 2.5 mm. The expansion contour was designed using a computer program based on the method of characteristics with the stipulation of uniform flow at the nozzle exit.[12] A time averaged image taken using a CCD camera of a cross section along the flow can be seen in figure 2.2. Correspondingly, a cross section across the flow can be seen in figure 2.3.

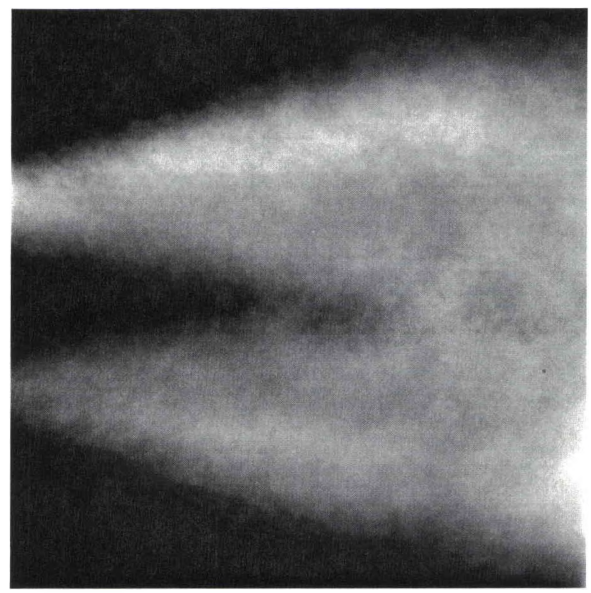

Figure 2.2 Cross Section of Jet Stream Along Flow[14]

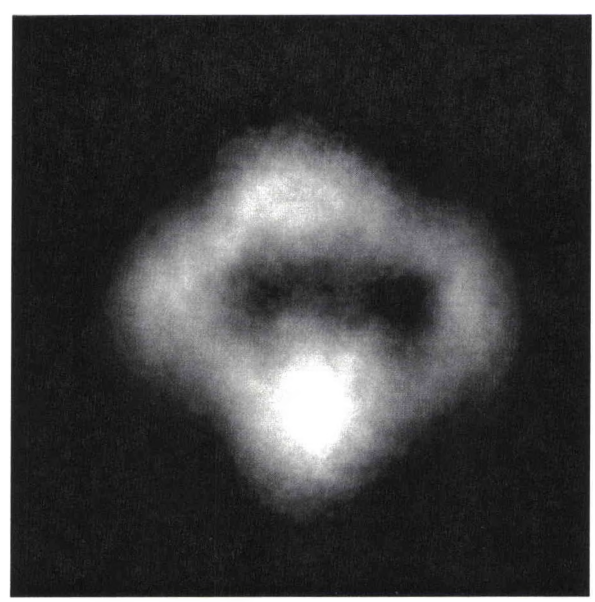

Figure 2.3 Cross Section of Jet Stream Across Flow[14]

The jet stream exhausts into a large room that can also serve as an anechoic chamber, and exits the facility through a large bell-mouth at the opposite end of the chamber. Pressure to the stagnation chamber is controlled manually through the actuation of a Fisher control valve and can be maintained at constant pressure within 0.1 psi.[15] Pressure was set for an ideally expanded flow and held constant through each set of experiments. A schematic of the anechoic chamber is given in figure 2.4 with a similar setup used in the recording experiment.

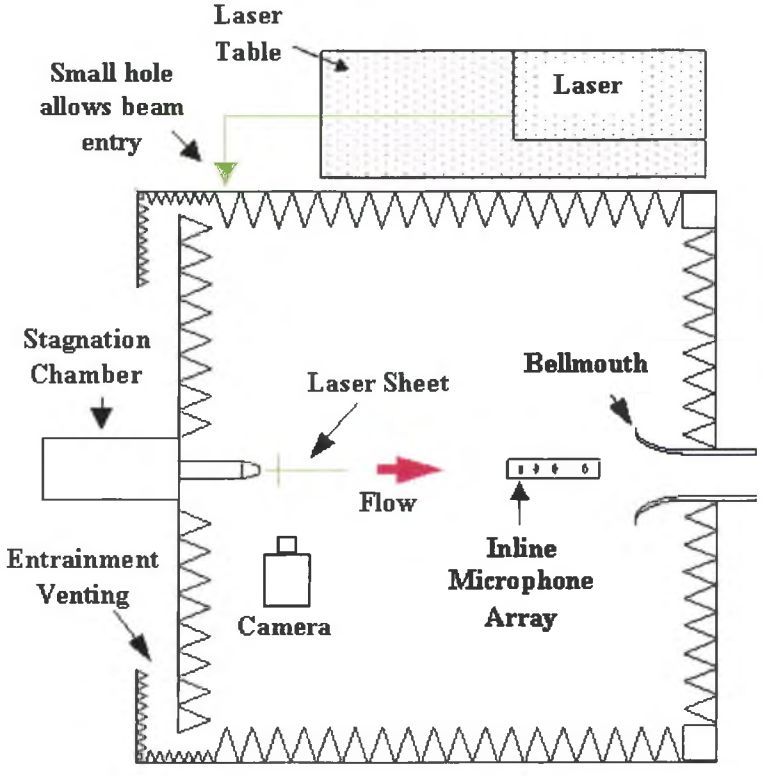

Figure 2.4 Schematic of GDTL Anechoic Chamber[15]

2.3 Flow Visualization

The next step is the recording of the images of the jet stream. For flow visualization, a pulsed, doubled, Nd:YAG laser beam was formed into a thin sheet and directed through a lane in the flow field. Laser light scattered from particles contained in the flow was then captured using a high speed CCD camera. Coinciding with the experimental setup is the analytical beam that will be discussed in the following section on obtaining the wavefront measurements. A diagram of the setup can be seen in figure 2.5. The CCD camera, while not visible in the figure, is positioned to record images represented by the dashed rectangle in the figure.

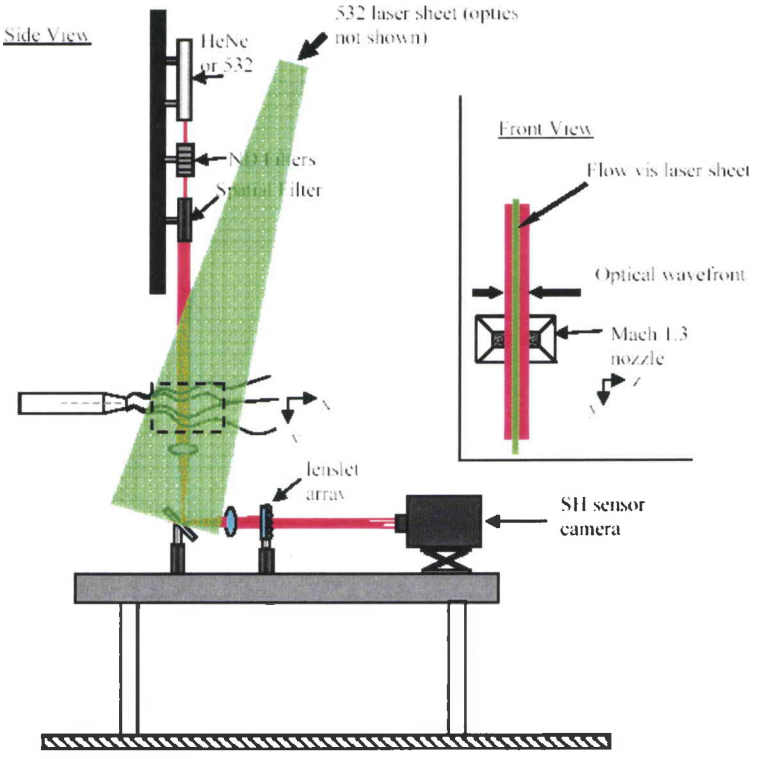

Figure 2.5 Setup for Flow Visualization and Wavefront Measurement[16]

In the recording experiment[17] performed at OSU's Gas Dynamics and Turbulence Laboratory in March 2003, seeding was provided using the product formation technique where water vapor contained in the warm moist ambient air condenses into nanometer-scale droplets upon entrainment into the jet stream and mixing with the cold, dry core air. Concerns about the size of the particles and their response time have been previously addressed by the OSU Research group, and the particles are believed to accurately depict the features of the flow.[18]

The advantage of this method of flow visualization is its simplicity and ability to mark the most dominant features within the mixing layer, in particular, the large-scale structures. An example of this is presented in figure 2.6. The technique gives a very good qualitative impression of the dynamics within the mixing layer. The disadvantage of the technique, however, is the limited amount of quantitative information that can be extracted.

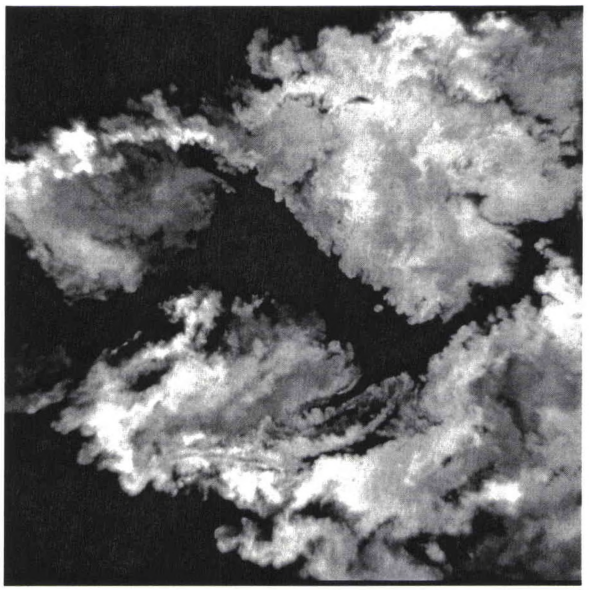

Figure 2.6 Instantaneous Flow Visualization Using High Speed CCD Camera[19]

Thurow, et al, explain the following in regards to the flow visualization.[16]

> The intensity of the scattered laser light is directly proportional to the number density and size of water droplets contained within a given volume of the flow. The number density and size of the particles, however, is a complex function of mixture fraction, density, temperature, and time. Due to this complex process, it is quite difficult to extract further information from images beyond the simple assessment that significant mixing has occurred where there is signal. Despite these limitations, the technique has been successfully used in a number of studies of compressible flows. In the context of aero-optics, there might be an additional concern about the influence of water particles on the optical wavefront passing through the mixing layer. The effect on the wavefront, however, is thought to be minimal as the particles are very small and the scattering is close to Rayleigh scattering regime. Furthermore, the particle number density is quite small compared to the number density of air molecules, i.e., less than one percent. Thus, the amount of light scattered towards the Shack-Hartmann sensor, mentioned in the following section, by the water particles will be orders of magnitude smaller than the light directly falling onto the sensor.

2.4 Simultaneous Wavefront Sensing

To obtain the wavefront information, a triggering system was used coinciding with the flow visualization to capture the wavefront information using a Shack Hartmann wavefront sensor. The particular Shack Hartmann sensor used for the corresponding experiments to this research was designed and built by Jeff Widiker et al.[20] While the following is a brief overview of the Shack Hartmann design, for a more in-depth analysis the reader is referred to Widiker's MS thesis.[10]

In the Shack Hartmann sensor, an optical wavefront is spatially sampled by a lenslet array. Each lenslet in the array produces its own focal spot on a recording medium. The lateral displacement of each of these spots from its reference location, as

previously produced by a reference wave-front, corresponds to the mean local wave-front tilt across that lenslet, as seen in figure 2.7.

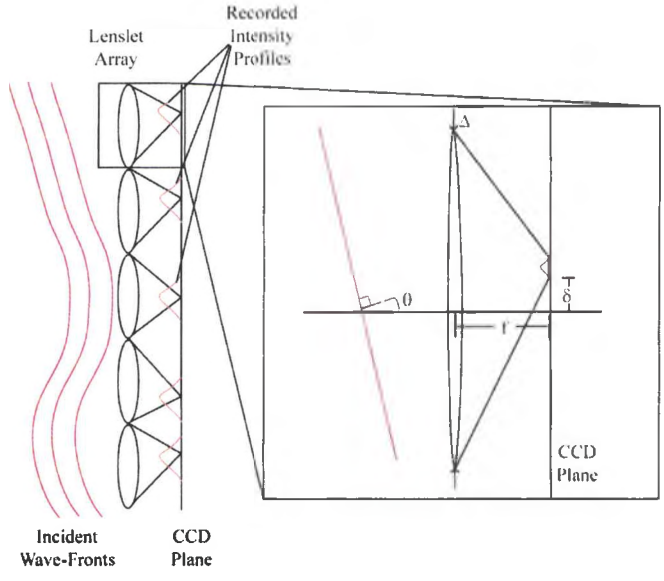

Figure 2.7 Schematic of Shack-Hartmann Wavefront Sensor[10]

Here, the angle θ, the wave-front tilt, is a function of spot displacement from a reference position, δ, related by the relationship of

$$\theta \approx \tan\theta = \frac{\delta}{f} \qquad (2.1)$$

where f is the focal length of the lenslet. For most Shack Harman sensors, in particularly high speed ones, θ is small. In order to produce an accurate wavefront measurement, the system must achieve sufficient sampling of the wavefront so that the spots on the recording medium are well defined. To match the spatial sampling criteria,

the lenslet pitch must be sufficiently small so that the wavefront can be well-represented by the resulting piecewise-linear approximation to the wavefront gradient. In addition, the amount of wavefront phase distortion the system can incur while still maintaining diffraction-limited performance is dependent on the types of aberration involved. This performance must satisfy the Rayleigh criterion of optical path difference, OPD, of less than a quarter wavelength across each subaperture.

In the absence of additional optics, the system's spatial sampling rate is determined solely by the lenslet array's pitch. If the system's spatial sampling rate is too low for a given aberration, meaning that the lenslet pitch is to large, the wave-front across each lenslet aperture no longer fulfills the linear approximation, causing spot distortions, known as spot tearing. Also, without additional optics, the spatial dimensions of the wave-front sampled by the sensor are dependent on the size of both the recording medium and the lenslet array. It is only the smaller of these that limits the amount of wavefront the sensor can sample.

Once the local wavefront gradients are calculated from the relative spot displacements, a reconstruction of the incident wavefront can be calculated. Since this reconstruction involves integration, the nominal phase of the wavefront is irretrievable without additional measurements. This wavefront measurement is made relative to a reference wavefront, which is typically planar. The use of a planar reference wavefront allows for the greatest dynamic range of the sensor since it places each spot's reference location directly behind the center of the corresponding lenslet.

CHAPTER 3

BEAM PROPAGATION

3.1 From Maxwell's Equations to the Wave Equation

Any valid discussion of beam propagation must begin with Maxwell's equations, presented here in the most commonly seen differential form.[21]

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \tag{3.1}$$

$$\nabla \times \vec{H} = \vec{J} + \frac{\partial \vec{D}}{\partial t} \tag{3.2}$$

$$\nabla \cdot \vec{D} = \rho \tag{3.3}$$

$$\nabla \cdot \vec{B} = 0 \tag{3.4}$$

Where $\vec{E}$ is the electric field

$\vec{H}$ is the magnetic field

$\vec{D}$ is the electric flux density

$\vec{B}$ is the magnetic flux density

$\vec{J}$ is the current density

ρ is the charge density

To begin working with these equations, we must look at the different assumptions and relationships of several parameters to obtain an understanding of how beam propagation occurs. The first step involves the constitutive relationships between the flux densities, $\vec{D}$ and $\vec{B}$, and the field amplitudes E and H. For linear, isotropic media, the relationships are given by equations 3.5 and 3.6.[22]

$$\vec{B} = \mu\vec{H} \tag{3.5}$$

$$\vec{D} = \varepsilon\vec{E} \tag{3.6}$$

Where μ is the magnetic permeability of the medium

ε is the electric permittivity of the medium

In addition, not only is the medium being used in this experiment linear and isotropic, it is also "source free" and nonconductive. Because of this, the current density, $\vec{J}$, as well as the charge density, ρ, can be assumed to be equal to zero. Using this assumption, as well as equations 3.5 and 3.6, substitution back into Maxwell's equations give:

$$\nabla \times \vec{E} = -\mu \frac{\partial \vec{H}}{\partial t} \tag{3.7}$$

$$\nabla \times \vec{H} = \varepsilon \frac{\partial \vec{E}}{\partial t} \tag{3.8}$$

$$\nabla \cdot \varepsilon\vec{E} = 0 \tag{3.9}$$

$$\nabla \cdot \mu\vec{H} = 0 \tag{3.10}$$

At this point equations 3.7 through 3.10 constitute a set of coupled, first order, partial differential equations for the electric and magnetic fields. The electric field is of more interest to us than the magnetic field. In order to eliminate B and H from the above equations, we take the curl on both sides of equation 3.11.

$$\nabla\times\left(\nabla\times\vec{E}\right)=\nabla\times\left(-\mu\frac{\partial\vec{H}}{\partial t}\right) \tag{3.11}$$

Using standard vector identities,[23] the curl of a curl can be expanded as

$$\nabla\times\left(\nabla\times\vec{F}\right)=\nabla\left(\nabla\cdot\vec{F}\right)-\nabla^2\vec{F}\,. \tag{3.12}$$

Note here that del squared operator, ∇^2, is the vector Laplacian operator acting on the vector $\vec{F}$. Using this identity, the left side of equation 3.11 now becomes

$$\nabla\times\left(\nabla\times\vec{E}\right)=\nabla\left(\nabla\cdot\vec{E}\right)-\nabla^2\vec{E}\,. \tag{3.13}$$

The final step deals with $\nabla\left(\nabla\cdot\vec{E}\right)$. Note that equation 3.9 has a similar expression, with exception of ε. We can expand 3.9 using the vector identity[23]

$$\nabla\cdot f\vec{F}=f\nabla\times\vec{F}+\nabla f\times\vec{F}\,. \tag{3.14}$$

Using the left hand side of 3.9 and 3.14 to solve for $\nabla \cdot \vec{E}$ gives

$$\nabla \cdot \vec{E} = -\vec{E} \cdot \frac{\nabla \varepsilon}{\varepsilon}. \tag{3.15}$$

Re-arranging the left side of equation 3.11 results in

$$\nabla \times \left(\nabla \times \vec{E}\right) = -\nabla\left(\vec{E} \cdot \frac{\nabla \varepsilon}{\varepsilon}\right) - \nabla^2 \vec{E}. \tag{3.16}$$

Next, we will work on the right side of equation of 3.11. Note that because the functions being dealt with are continuous, the order of the derivative operator and the curl can are reversible. Also, the magnetic permeability, μ, is independent of both time and space and can be brought in front of the curl. As a result, the right side of equation 3.11 turns into

$$\nabla \times \left(-\mu \frac{\partial \vec{H}}{\partial t}\right) = -\mu \frac{\partial}{\partial t}\left(\nabla \times \vec{H}\right). \tag{3.17}$$

Equation 3.8 can now be substituted in to give

$$\nabla \times \left(-\mu \frac{\partial \vec{H}}{\partial t}\right) = -\mu \frac{\partial}{\partial t}\left(\varepsilon \frac{\partial \vec{E}}{\partial t}\right). \tag{3.18}$$

The right side of equation 3.11 then becomes

$$\nabla\times\left(-\mu\frac{\partial\vec{H}}{\partial t}\right)=-\mu\varepsilon\frac{\partial^2\vec{E}}{\partial t^2}, \tag{3.19}$$

and the overall form equation of 3.11 then becomes

$$-\nabla\left(\vec{E}\cdot\frac{\nabla\varepsilon}{\varepsilon}\right)-\nabla^2\vec{E}=-\mu\varepsilon\frac{\partial^2\vec{E}}{\partial t^2}, \tag{3.20}$$

which can be rearranged to a final result of

$$\nabla^2\vec{E}-\mu\varepsilon\frac{\partial^2\vec{E}}{\partial t^2}=-\nabla\left(\vec{E}\cdot\frac{\nabla\varepsilon}{\varepsilon}\right). \tag{3.21}$$

While equation 3.21 looks very similar to the wave equation one commonly sees in textbooks, there is still the extra factor on the right side to deal with. This term will be non-zero when there is a gradient in the permittivity, and while such gradients are rare except in waveguides with strong index contrasts, for most structures, especially in free space, the term is negligibly small.[24] Because of this, we can assume this part of the equation to be zero, which results in

$$\nabla^2\vec{E}-\mu\varepsilon\frac{\partial^2\vec{E}}{\partial t^2}=0. \tag{3.22}$$

This looks very much like the classical traveling wave equation[25]:

$$\nabla^2 f - \frac{1}{v^2}\frac{\partial^2 f}{\partial t^2} = 0. \qquad (3.23)$$

Noting that v is the speed of the traveling wave, from equation 3.22 we can deduce that the speed of light in the medium is

$$v = \frac{1}{\sqrt{\varepsilon\mu}}. \qquad (3.24)$$

Substituting this back in to equation 3.22 gives the final result of

$$\nabla^2 \vec{E} - \frac{1}{v^2}\frac{\partial^2 \vec{E}}{\partial t^2} = 0. \qquad (3.25)$$

This is Maxwell's wave equation in a linear non-dispersive source-free medium.

3.2 Solving the Wave Equation

While there are many methods of solving the wave the equation the method that will be used here is that of the Fourier transform method. The first assumption, made for simplification of the mathematical calculations, is that consideration will be given to only a single spatial direction, in this case designated as the $\hat{z}$ direction, or the propagation

direction. As a result, the vector Laplacian operator, ∇^2, given in equation 3.26[27] is then reduced to the term in only the $\hat{z}$ direction as

$$\nabla^2 = \frac{\partial^2}{\partial x^2}\hat{x} + \frac{\partial^2}{\partial y^2}\hat{y} + \frac{\partial^2}{\partial z^2}\hat{z}. \qquad (3.26)$$

Using this assumption, equation 3.5 can then be rewritten with E varying only in the $\hat{z}$ direction, in the form of

$$\frac{\partial^2 E(z,t)}{\partial z^2} = \frac{1}{v^2}\frac{\partial^2 E(z,t)}{\partial t^2}. \qquad (3.27)$$

Next, we will assume that the medium is linear and nondispersive. As a result, the Fourier transform, seen in equation 3.28, can be applied to both sides of equation 3.27, resulting in equation 3.29. In the following calculations, the integration limits are from -∞ to +∞ unless otherwise noted.

$$f(x) = \int F(k)e^{ikx}dk \qquad (3.28)$$

$$\int \frac{\partial^2 E(z,t)}{\partial z^2} e^{-ikz}dz = \frac{1}{v^2}\int \frac{\partial^2 E(z,t)}{\partial t^2} e^{-ikz}dz \qquad (3.29)$$

Note in equation 3.29, the appearance of the parameter k. This is the spatial frequency which arises from the Fourier transformation and it represents the well known propagation constant or wave number.

The Fourier transformed field will be written as:

$$\psi(k,t) = \int E(z,t)e^{-ikz}dz\,. \tag{3.30}$$

Substituting this in equation 3.29 results in

$$(ik)^2\psi(k,t) = \frac{1}{v^2}\frac{\partial^2\psi(k,t)}{\partial t^2}. \tag{3.31}$$

The solution of 3.31 can be written as:

$$\psi(k,t) = A(k)e^{-ikvt} + B(k)e^{ikvt} \tag{3.32}$$

In equation 3.32, the coefficients A and B are functions of k as well as the boundary conditions. Furthermore, the two parts in equation 3.32 represent waves propagating in the positive and negative directions respectively. These two waves are coupled through reflective processes in the medium. If the environment does not contain any strongly reflective surfaces, or large refractive index contrasts, we can ignore the coupling between these two counter-propagating waves. Propagation through air is an ideal example where this approximation is very reasonable. As a result, we will consider only the forward propagating field and ignore the backward propagating field. Because of this, equation 3.32 can be rewritten as

$$\psi(k,t) = A(k)e^{+ikvt} \tag{3.33}$$

Now, using equation 3.33, the inverse Fourier transform is taken of equation 3.31 to give:

$$E(z,t) = \int \psi(k,t)e^{-ikz}dk \tag{3.34}$$

$$E(z,t) = \int [A(k)e^{+ikvt}]e^{-ikz}dk \tag{3.35}$$

$$E(z,t) = \int A(k)e^{-ikvt}e^{-ikz}dk \tag{3.36}$$

$$E(z,t) = \int A(k)e^{-ik(z+vt)}dk \tag{3.37}$$

3.3 Significance of the Wave Equation Solution

Basically, the solution gives an oscillatory function that has a sinusoidal variation with either z or t. If either z or t is set to a constant, the resultant is a sinusoidal varying function of the form e^{-ikz} or e^{-ikvt}, that is frozen in time or space accordingly. With either case, it is the real part of the function that is used to obtain the value of the amplitude. The reason this is significant for this research is because what is really being applied is that the initial amplitude of the wave is being manipulated by a change in phase, where the phase factor, ϕ, is denoted by

$$\phi = k(z - vt) \tag{3.38}$$

Also, the angular frequency, ω, is defined as the propagation constant, k, times the wave velocity, v, written as

$$\omega = kv \tag{3.39}$$

Expanding out equation 3.38 and using equation 3.39 results in the phase written in the commonly seen form of

$$\phi = kz - \omega t \,. \tag{3.40}$$

By considering the value of the wave for some constant phase, one can now track how the wave moves with time, since the amplitude of the wave function remains constant when the argument of the wave function is constant.

3.3.1 Simulation of Beam Propagation in Flow

The emphasis is to simulate a laser beam passing through a high speed flow, a condition similar to what is represented in figure 3.1.

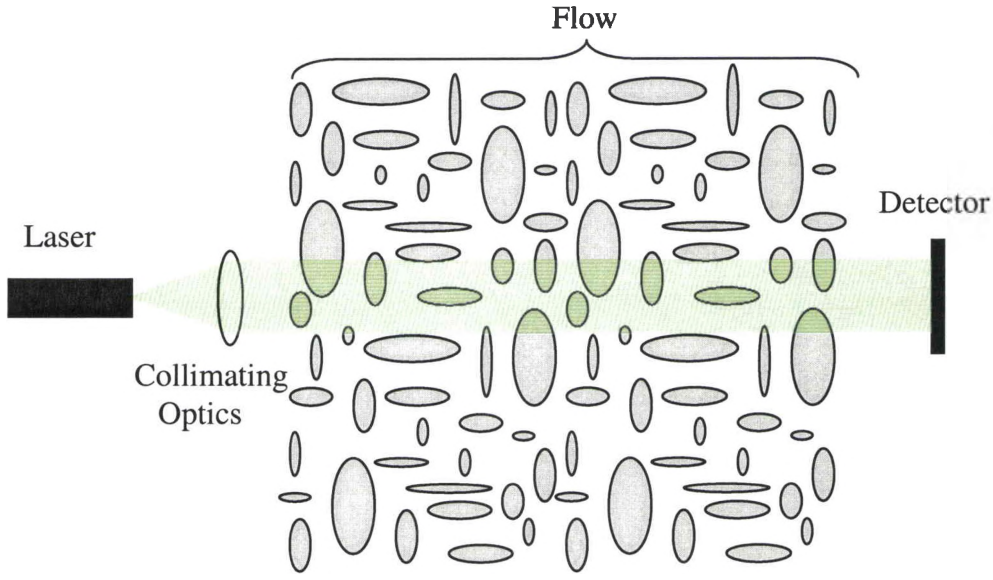

Figure 3.1 Basic Concept Laser Beam Through Flow

Basically the beam is collimated and passes through a high speed flow to a detector. This detector, as described in chapter 2, measures the wavefront. The actual physics behind figure 3.1 is as follows. The high speed flow is made of a variety of different eddies, or turbules, mixing up the high speed jet core and the ambient air. While there are several methods for modeling the actual flow characteristics[29,30], which will not be discussed here, it comes down to the basic idea that there are several small turbules, as represented in figure 3.1 by the various ovals, that are homogenous, i.e. contain the same refractive index, pressure, temperature, etc., that are dependent on several variables, including the velocity of the stream. While the simulation will be using the flow data from the recording experiment[17] and will be described in following chapters, the methodology of representing the flow will be discussed here.

Recall from the previous paragraph that each turbule has a homogenous refractive index. If one goes back to the use of Snell's law and drawing a ray diagram, figure 3.2 can be redrawn into a ray trace version seen in figure 3.2 as the propagation geometry.

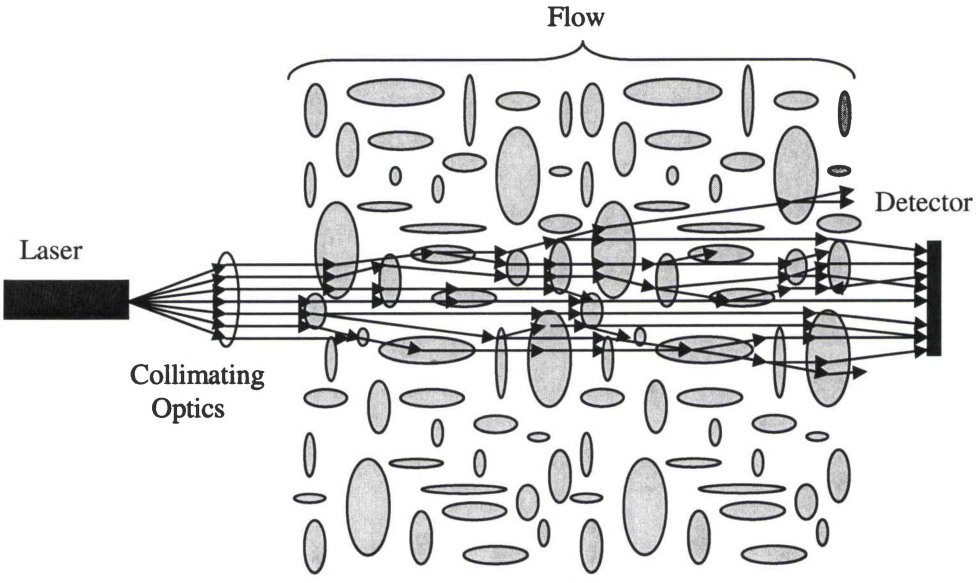

Figure 3.2 Propagation Geometry for Beam Propagation Through Flow

The reason for going into such a deep discussion of solving the wave equation for light and then nonchalantly jumping over to ray tracing becomes apparent here. While the model being developed here is not going to model the flow, the change of phase to the beam passing through the flow is the key aspect to the model. Without a change of phase, the beam would be unchanged. So this model must account for the phase. To do this, the phase must be related to the parameters of the propagated beam and the flow through which it is passing. This relationship comes in the form relating the refractive index to the velocity of the light in the medium, seen in equation 3.41[31] and the wavelength of the laser beam, as seen in equation 3.42.[31]

$$n = \frac{c}{v} \tag{3.41}$$

Where n is the refractive index

c is the speed of light in a vacuum

v is the speed of light in a medium

$$k = \frac{2\pi}{\lambda} \tag{3.42}$$

Where k is the propagation constant

λ is the wavelength

These equations now give a way to simulate the conditions of the flow, given that the refractive index and laser beam wavelength are known. As a result, one can go back to equation 3.40 and, along with the known variables and equations 3.38 through 3.42, can come up with an equation which can be coded into a program for the beam propagation. However, the solution is not as easy as just plugging in the known parameters. For example, if the distance traveled is large, or the change in refractive index of the medium is large, in real world conditions, the beam will fluctuate to a great degree over the propagated distance. As a result, the appropriate method for modeling needs to be considered.

If one goes about modeling the situation as shown in figure 3.2, it is known as using an extended medium model.[32] If the medium exists everywhere along the propagation path between the transmitter and the receiver, and is modeled as having the medium along the entire path, this is considered to be an extended medium model. While this model does overcome the issue of large distances and changes in refractive index, it would require a lot of processing power and time to take in account each step of the propagation. There is however, a simpler method that uses less processing.

This easier method of modeling is the approach of using a phase screen model.[32] Basically what this method of modeling is doing is compressing the phase information of a particular condition or situation into a single phase screen. This phase information will include the distance traveled as well as the change of refractive index of the medium. This approach, seen in figure 3.3, has been used for many years to study scintillation phenomena. In the figure, a beam is propagated over a distance of L_1, up to the phase screen, then propagated over a very short distance, L_2, over the phase screen, and the propagated over a distance, L_3, to reach the detector. The actual physical distance of L_2 is very small in comparison to L_1 and L_3, usually a distance of one propagation unit or less, and often times given no value at all because the phase screen already contains the distance information and is simply plugged into the propagated beam.

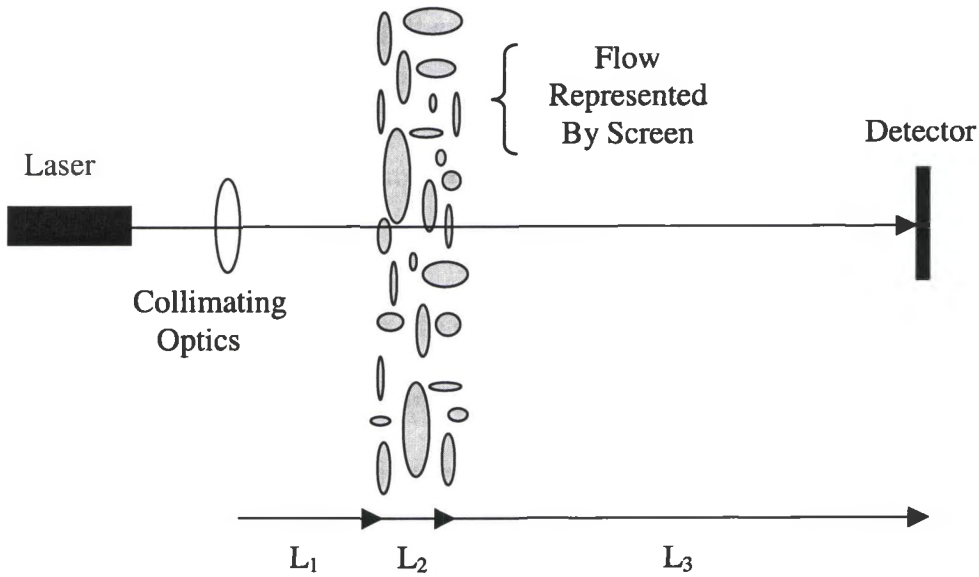

Figure 3.3 Phase Screen Model Visualization

However, as mentioned previously, if the distance traveled is sufficiently large or the change of refractive index is large, aberrations can be introduced over the propagated distance. To account for this, multiple screens can be used as a replacement for the single screen. This approach is called, as one would expect, the multiple phase screen model[32]. A visualization of this method can be seen in figure 3.4. While the figure here involves the use of two screens, this approach can use as many screens as necessary to adjust for the developing aberrations over the distance propagated.

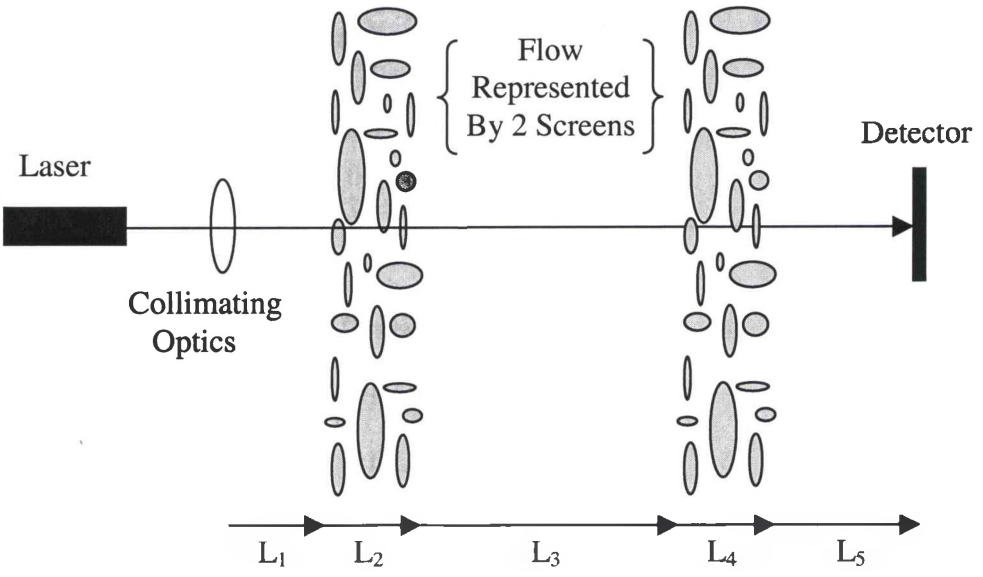

Figure 3.4 Multiple Phase Screen Model Visualization

In figure 3.4, the lengths of screens, L_2 and L_4, are similar to the length of single phase screen model. However, the distance between the screens needs to be related to the actual distance of the flow by a ratio related to the distance propagated. For example, if the flow were 100 meters in length, and the each phase screen represented 50 meters, then the length L_3 needs to be an appropriate portion of the actual 100 meters. The most common method of determining the appropriate distance is to divide the distance represented by the phase screen by one more than the number of phase screens. This allows for half of the propagation to occur before and half of the propagation to occur after each of the represented screens. The reason that the distance needs to be simulated here is that the aberrations, shift for example, need distance to develop before entering in the next phase screen information. By allowing for the ratio between the multiple screens, the model can allow for equal distances between adjoining screens.

3.3.2 Beam Propagation Methods

Once the phase screen is computed, it can be substituted into the wave equation and then a computer can be used to do the integration process. As the distances become large and/or as the phase becomes more and more complicated, the direct integration method would need a lot of processing power, even for the single screen method. Several methods can be applied to simplify this calculation. Among them are: the fast Fourier transform (FFT), convolution, and finite differences (FD). The method used for the simulation is dependent on the type of software being used. The software used for this research will be explained further in chapter 5. It will be there, that the actual methods of propagation, whether FFT, finite differencing, or direct integration, will be revealed and given proper explanation. For the time being, it is sufficient enough here just to list the methods of beam propagation.

CHAPTER 4

CALCULATION OF PHASE

4.1 False Temperature Maps

While the jet stream images from the recording experiment[17] are not true maps of the refractive index or the density of the stream, they are, however, useful in determining characteristics of the stream. As described in chapter 2, in order to obtain the air speed involved in the experiment, the air is filtered, dried, and pumped into the large tanks. As a result, the jet stream released from the tanks is made up of very cold, very dry air. As the jet stream meets with the warmer, moister, stagnant air, there is a mixing of the two. Ultimately the stream slows down, warms up, and becomes part of the stagnant air. This effect not only occurs downstream, but it also occurs around the outsides of the jet stream. Because of this, and if one treats air as an ideal gas, one can go back to the relationship between stagnation temperature, flowing temperature, and Mach number for a flowing ideal gas. When brought adiabatically to rest, the kinetic energy of the stream is converted into enthalpy. As a result, equation 4.1[33] can be used.

$$\frac{T_a}{T_s} = 1 + \frac{\gamma - 1}{2} M^2 \qquad (4.1)$$

Where T_a is the absolute temperature of the stagnant, ambient air

T_s is the absolute temperature of the jet stream

M is the mach number of the speed of the air

γ is the specific heat of the gas being analyzed.

Given the parameters from the testing conditions measured and calculated by Brian Thurow and Jeff Widiker at the recording experiment[17] of T_a = 286 ° K, M = 1.28, and treating air as an ideal gas, γ = 1.4, the temperature of the stream, T_s, is calculated to be 215.4 ° K. With the temperature of the stream well below the dew point, in the areas where the stream and ambient layers mixed, water droplets and ice formed from condensation and the cold temperature. While these particles do produce a significant return from the laser beam, it was concluded by Thurow et al, that due to the dynamic range of the detector used in the recording experiment of 5000:1, the signals obtained in the images mark a very large and significant portion of the mixing layer and that the size and shape of structures may be inferred from the images.[34]

Because of this, the images can be seen as a temperature contour map with fairly coarse boundaries. Seen in figure 4.1, the mixing layers give a false representation of temperature by scattering light off of the condensation where the accumulation of moisture is most abundant. Although the core of the stream is at a much cooler temperature, because it is dry and does not contain particles to scatter the light, it looks like the outside ambient air.

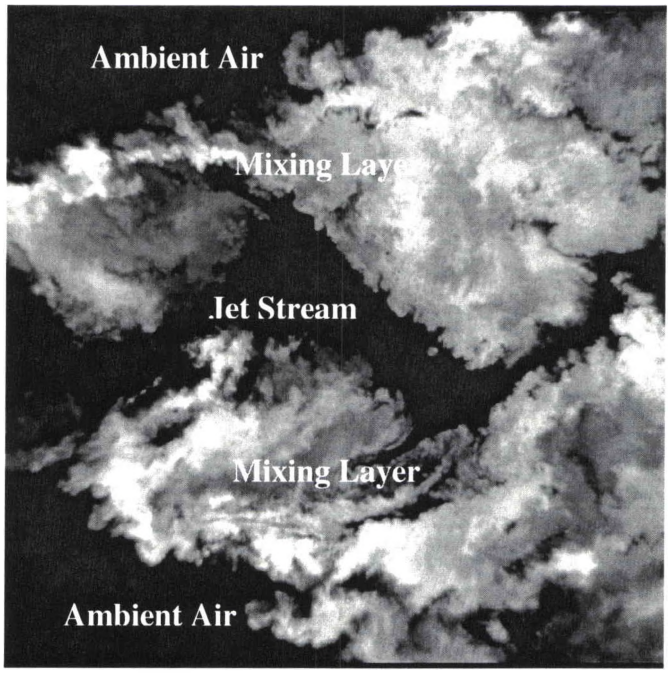

Figure 4.1 OSU Image – "Temperature Map"

While the outside ambient air and jet stream can now be solidly defined in terms of temperature, the mixing layer can not. The determination on how the mixing layer is to be analyzed will be discussed later, but for now it can be only defined by boundary.

4.2. Calculation of Refractive Index

The next step is to convert the above previously mentioned "temperature maps" into useful maps of refractive indices so as to calculate the phase. Here, however, a brief discussion is needed to fully understand the process in performing this endeavor.

The definition of the ratio of the speed of light in a vacuum, c, to the speed of light in a given material, v:

$$n = \frac{c}{v}. \tag{4.2}$$

The formulas for c and v are given by equations 4.3[35] and 4.4.[35] as:

$$c = \frac{1}{\sqrt{\varepsilon_o \mu_o}} \tag{4.3}$$

$$v = \frac{1}{\sqrt{\varepsilon_r \varepsilon_o \mu_r \mu_o}}, \tag{4.4}$$

where ε_o is the permittivity of free space of $8.885 \cdot 10^{-12} C^2 / N \cdot m^2$

μ_o is the permeability of free space of $4\pi \cdot 10^{-7} N / A$

ε_r is the relative permittivity for the material

μ_r is the relative permeability for the material.

In the experiment, however, the material being used is a gas, in this case, air. As a result, the following assumption can be made. Commonly in the case of gases, being for the most part non-magnetic, μ_r is close to unity. As a result, we can get:

$$n = \frac{c}{v} \Rightarrow \frac{\frac{1}{\sqrt{\varepsilon_o \mu_o}}}{\frac{1}{\sqrt{\varepsilon_r \varepsilon_o \mu_r \mu_o}}} \Rightarrow \frac{\sqrt{\varepsilon_r} \cdot \sqrt{\varepsilon_o \mu_o}}{\sqrt{\varepsilon_o \mu_o}} \Rightarrow n = \sqrt{\varepsilon_r} \tag{4.5}$$

This relationship holds true only for gases that are non-magnetic, or non-polar, for example, such as nitrogen, oxygen, helium, neon, etc., but it begins to break down for

polar gases, such as ammonia and water vapor.[36] This assumption is key because the gas used in the experiment is air. The air, being composed mostly of nitrogen, oxygen, and argon, is dried before being stored in the compression tanks, described previously chapter 2. As a result, equation 4.5 is valid for the jet stream. However, while there is water vapor in the ambient and mixing layers, which could introduce error into this assumption, it will be shown later that due to the short distances traveled, the error is relatively small and equation 4.5 still remains valid.

Next equation 4.6[37] is used for finding the permittivity of the gas by

$$\varepsilon_r = 1 + \frac{\eta\alpha}{\varepsilon_o}, \tag{4.6}$$

where ε_o is the permittivity of free space of $8.885 \cdot 10^{-12} C^2 / N \cdot m^2$

α is the polarizibility of the gas

η is the number density of the gas given by equation 4.7[37]

$$\eta = \frac{P}{k_B T}, \tag{4.7}$$

where P = the pressure of the gas in N / m^2

T = the absolute temperature of the gas in K

k_B = the Boltzmann constant of $1.381 \cdot 10^{-23} J / K$.

The number density is calculated from the pressure measured at the site of experiment, the measured temperature for the ambient air, the calculated temperature from equation 4.1 for the jet stream, and the Boltzmann constant to give values of η_{amb} = 2.45913 m^{-3} for the ambient air and η_{core} = 3.26494 m^{-3} for the jet stream. These numbers are then used in equation 4.6 to calculate the permittivity of the air.

One aspect of equation 4.6 that needs to be determined however, is the polarizability. The following table, table 4.1, gives the composition of air in percentages as well as the corresponding polarizabilities from the CRC Handbook of Chemistry and Physics.

Table 4.1 Composition of Air and Corresponding Polarizabilities

Gas	Composition of air[38] %	Polarizability[39] $\cdot 10^{-40}$ F^{-1} m^4
Nitrogen (N_2)	78.03000	1.7403
Oxygen (O_2)	20.93000	1.5812
Argon (Ar)	0.94000	1.6411
Carbon Dioxide (CO_2)	0.03300	2.9110
Neon (Ne)	0.00150	0.3956
Helium (He)	0.00052	0.2049
Krypton (Kr)	0.00014	2.4844
Xenon (Xe)	0.00001	4.0440

Taking the weighted values of each of the gases from table 4.1 and adding together the polarizabilities gives an overall polarizability of air to be 1.70530$\cdot 10^{-40}$ F^{-1} m^4. Now, having all of the variables defined, the polarizabilities of air is then substituted

into equation 4.6 along with the number density to give the permittivity of the ambient air, ε_{ramb}= 1.0004739 and the permittivity of the jet stream, ε_{rcore}= 1.0006292, resulting in refractive indices of n_{amb} = 1.0002368 and n_{core} = 1.0003143.

The final step in determining the refractive index map is to determine the magnitude of the mixing layer's refractive index. The original intent was for the OSU group to characterize a three dimensional flow and measure its density profile. However, due to limited resources, the work performed only entailed a two dimensional image of the flow, as described in chapter 2. As a result, a full analysis of the mixing layer could not performed. For simulation purposes, this meant that certain approximations had to be made for the mixing layer. It was determined that the mixing layer could be modeled reasonably well by taking the average of the core and ambient layer indices. The results of this simulation would be compared to the measurements for validity.

As a result, the average of the core and ambient layers was used in the simulation giving the mixing layer a refractive index of n_{mix} = 1.0002757. A sample image showing the flow with corresponding refractive indices values is given in figure 4.2

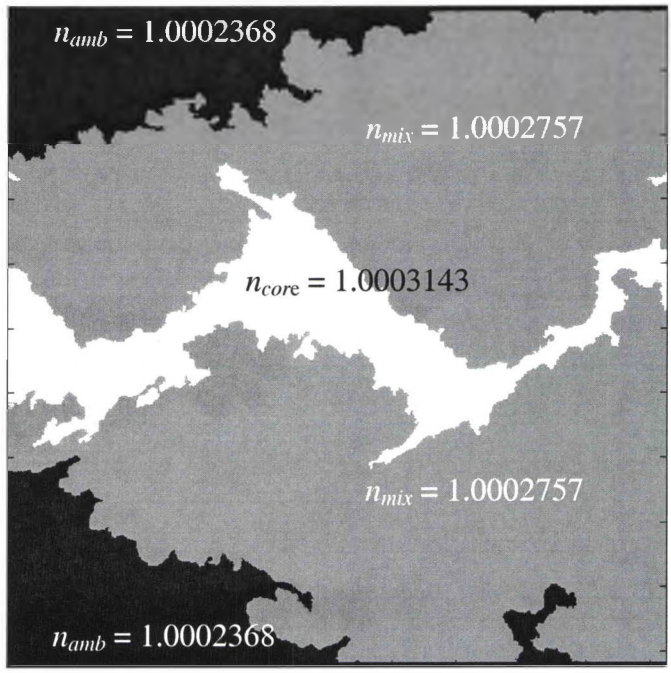

Figure 4.2 Refractive Index Map

4.3 Choice of Turbulence Modeling Method

Recall from chapter 3, that the flow can be modeled in many different ways. In this research, the data maps obtained by recording the laser beam propagation through the jet stream represent cross sections of the jet stream, as described in chapter 2. The decision was made to represent the jet stream first by a single phase screen model for this situation. If the analysis proved this could not be accomplished because too much phase information was trying to be assumed, i.e. meaning that the beam was changing dramatically in a short propagation distance, then the simulation would have to be changed to incorporate two or more phase screens. The conditions for which this would

arise are explained in the analysis of the difference phase propagation methods in section 6.2.5

The factors that make the simulation of the flow using one screen a better decision over the extended medium model are; the propagation distance is relatively short, the degree with which the phase changes is assumed to be relatively small, and ease of implementation of this type of phase screen. While the use of multiple screens would be more accurate, as mentioned, it requires extra calculation steps and processing time. This method was tested by breaking up the phase screen into four layers, however, it yielded no improvement in the accuracy of the results. This was assumed to arise from the fact that the total distance traveled across the jet stream was relatively small and that the changes in the refractive index of the model were also small, resulting in smaller amounts of aberration. The analysis indeed proved this, as discussed in section 6.

4.4 Calculating Phase from Refractive Index

The calculation of the phase from the refractive index begins first with the calculation of the optical path length, Γ. In an inhomogeneous medium where the refractive index is a function of position, the optical path length is given by equation 4.8.[40]

$$\Gamma = \int n(x)dx \tag{4.8}$$

However, the refractive index map shown in figure 4.2 is a pixilated image that gives a distinct value of refractive index for each individual pixel. Because of this, along

with the assumption that due to the small changes in refractive index that the resulting deflection angles will be small, i.e. much less than one degree, we can break the integral into a sum as seen figure 4.3 and represented in equation 4.9. Note in this figure that the beam propagation would be vertical from top to bottom, denoted by the refractive index numeration.

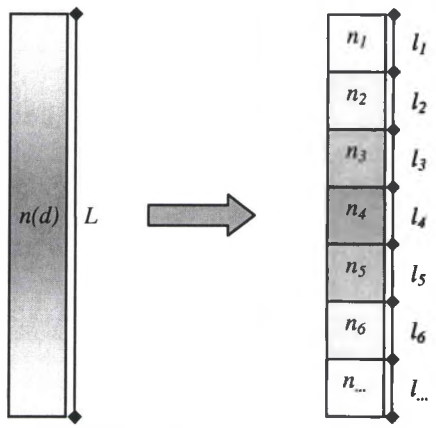

Figure 4.3 Pixilization of Refractive Index

$$\Gamma = \sum_{i=1}^{L} n_i l_i \qquad (4.9).$$

More than the absolute optical path, it is often more useful to find the optical path difference, δ, and use this in the relationship to find the change of phase. The optical path difference is give by the formula in equation 4.10 of

$$\delta = \Gamma - \bar{\Gamma}, \qquad (4.10)$$

where $\bar{\Gamma}$ represents the average value of the optical path difference. This is now substituted into the equation for finding the change of phase, $\Delta\phi$, equation 4.11, dependent not only the optical path difference, but also on the wavelength, λ, and given by:

$$\Delta\phi = 2\pi\frac{\delta}{\lambda}. \qquad (4.11)$$

Note that the direction of propagation of the laser beam through the flow is down through the beam, as shown here, in figure 4.4.

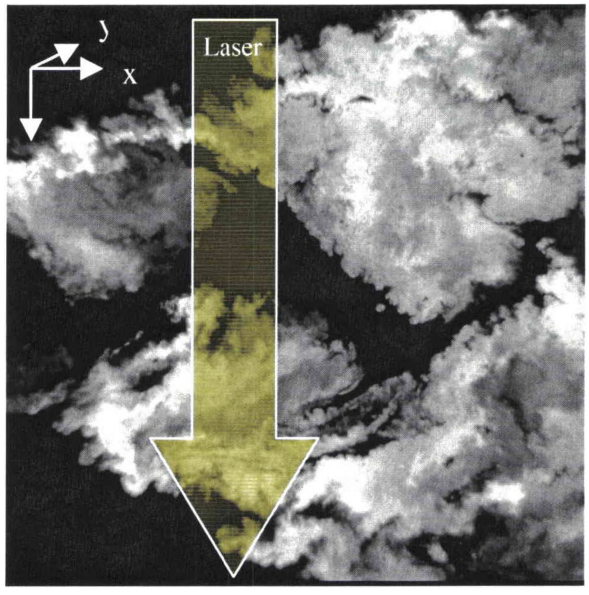

Figure 4.4 Laser Beam Propagation Axis Designation

After the image is transformed into a map of refractive indices, it is of significance to note that what is being defined in reality is a pixilated image where each pixel is now denoted as a uniform refractive index. By knowing that the total physical

screen size is two and five-eighths of an inch tall and that the image has a resolution of 1024 x 1024 pixels, one can now find the length, l_i, of each pixel, and perform the optical length summation in the z direction. An example of the calculated phase screen can be seen in figure 4.4 below. Here the image given in figure 4.1 was used as the input and the process for phase calculation outlined in this section was followed. The degree of phase change is denoted in the vertical axis while the distance across the image is represented in the horizontal axis.

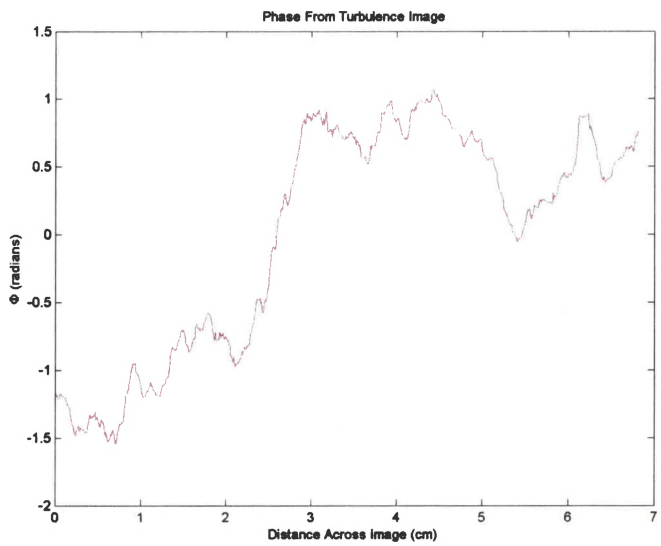

Figure 4.5 Calculated Phase From Turbulence Image

One must remember, however, that the phase being calculated is only one small slice of the entire stream, as discussed earlier. It is here, due to lack of substantial information pertaining to the rest of the jet stream, that we must make a limitation of the model. The calculated phase is converted into a three dimensional cube of phase using the phase screen represented in the x-z plane and expanding the same plane to all steps of

the y axis, and then compressed the entire volume into a single phase screen in the x-y plane that contains the three dimensional phase information. In doing so, the two dimensional phase is imposed across the entire screen. Because of this, upon comparison with the measured phase, only one line across of the measured phase will match up with the prediction model. Error is also introduced here due to the fact that there is a difference between the wavefront sensor size as compared to the slice size. This error, however, will be discussed further in chapter 5.

CHAPTER 5

STUDY OF AVAILABLE BEAM PROPAGATION SOFTWARE

5.1 Initial Investigations

The first task in the simulation and prediction of a laser beam through a flow was to determine what types of software were available to aid in the propagation. Both commercial and shareware/freeware were considered as potential options. However, the first findings of freeware and shareware were more of partial codes written in different languages. Because of the time and effort it would take to further rewrite these codes to make them "talk" to each other, it was decided that commercial software was going to be the best available option. The cost of the commercial software would also be a factor. While there was budget for the program, it was not unlimited, so software cost needed to be very closely evaluated to ensure that the final decision would allow optimal flexibility at the lowest cost. As a result, the following programs were identified as possible beam propagation candidates

5.2 ABCD Gaussian Beam Propagation Software

ABCD Gaussian Beam Propagation Software[41], developed by KAGI, Inc.[42] in Berkley, California, propagates a Gaussian laser beam through standard optical elements. These elements include free space, a dielectric, thin lens, curved mirrors, Gaussian ducts,

and dielectric interfaces. Basically this software uses the application of ABCD matrices, a.k.a. Ray matrices, hence the name of the program, with the multiplication of 2 x 2 matrices and its application of a user-defined complex radius. This stand alone program uses a point-and-click interface connected to an engine doing standard ABCD matrices & complex radius calculations.

The pros of this program is that it had a user friendly interface for setting up the initial beam parameters, i.e., wavelength, beam diameter, divergence, etc., it simulates a Gaussian laser beam, which is the assumption being used in this project, it allows for matrix manipulation, and it is shareware for a cost of $10.00. It is a nice program intended for use as a research tool for people working in optics labs.

However, it is limited in being able to handle complex structures, for example, the determined phase screen mentioned in chapter 4. Also, this software is available for Macintosh only. It doesn't allow for much manipulation or exploration of the beam downstream other than simple plots.

5.3 WinLase

WinLase[43], designed by Laser & Computer Consulting[44] and released through Future Laser Technologies[45], is a software package for designing laser cavities and for analyzing the propagation of Gaussian beams in complex optical systems. This program uses matrix algorithms for its calculations, similar to the program mentioned in the previous section. This was developed in C++ code for MS Windows based computer systems. User friendly interface allows plugging in optical elements with simple point & click operations. Laser cavities and single-pass systems can be analyzed simultaneously,

thus enabling the design of a laser and its pump optics in one simulation. The results can be displayed in a variety of plots.

The pros of this program include the use of Gaussian beam propagation, easier application of putting in complex optical elements, though with some work involved, and beam output plots. However, this program focuses more on laser cavity design, which is not the object of this research. While there are several options for plotting beam output, porting the information to other programs was considered to be an important issue. Also, the price of $895.00 while acceptable for a cavity design program, was not favorable enough for the projected uses in this research.

5.4 SCIOPT Software

SCIOPT Enterprises[46] offers a wide array of Optics software tools and high performance optical instruments. Among the software side are programs such as OPTEC-IV[47] for general optical system design of conventional and laser based systems, SIGRAPH-OPTIK[48] for integrated optics applications, and PARAXIA and PARAXIA-Plus[49] for laser system design, laser resonator design, and beam propagation. Of these, the last two, PARAXIA and PARAXIA-Plus were looked at as possible candidates for the beam propagation code. Basically PARAXIA-Plus is an extended version of PARAXIA and will be the center of discussion for this group of models.

PARAXIA-Plus is a stand alone module available on several platforms. The software uses optimized algorithms including the Fast Fourier and Fast Hankle transforms for beam propagation analysis. Propagations with up to 128,000 sampling points are possible with this software. The propagation engine can accept user supplied

code to customize wavefront operations and gain sheets. User generated scripting of complex operations is also supported. Several optimization tools and access to laser, glass, and catalog element libraries are provided in the model. The propagated beam can be analyzed at any point and the program contains automatic M^2 and tolerance analysis calculations.

The advantages of this program include the use of Gaussian beam propagation, the ability to customize the wavefront operations, and the ability to look at the beam at any point along the propagation. The program contains extensive databases for lasers, glass and other optical materials, hence a high cost of $2495.00. Although the program can be used to simulate the jet stream and for beam propagation, the extra features (and cost) of the program was not deemed necessary for this research.

As a brief side note, it also of importance to mention here that other powerful software, such as Code V and Zeemax, while superb for lens design and ray tracing applications through a system, where not included in this study. The reasoning behind this is that these types of software are specialized more for system design and implementation as well as come prepackaged in the beam propagation techniques. The software being used for this research needs to be as much open source as possible to be able to explore, understand, and make modifications if necessary, to the processes occurring in the propagation through the high speed flow. In order to obtain a better understanding of the research, reliance too heavily on one particular program without being able to obtain the underlying coding would provide less benefit in the learning process than programs that have the entire coding structure revealed.

5.5 WaveTrain

WaveTrain[50] is commercial software produced by MZA Associates Corporation[51], originally designed for the Airborne Laser Program[52] for the Air Force Research Laboratory Directed Energy Directorate.[53] WaveTrain is a powerful and flexible tool for modeling optical systems, especially systems that use adaptive optics to counteract the effects of atmospheric turbulence, such as laser weapons systems and compensating imaging systems. It combines diffraction optics modeling techniques and an advanced software architecture and is highly reconfigurable. The various optical components are represented as "blocks", similar to the Simulink[54] structure. Optical system models are assembled by connecting the inputs of each block to the outputs of others. The model differs from the typical Simulink form in that input/output connections can represent complex interactions, such as optical interfaces.

WaveTrain uses specialized graphical user interfaces for setting up system parameters and can setup complex systems such as the deformable mirror actuator geometry and an automated facility to help set up studies that could possibly contain hundreds of runs and automatically manages all the output. WaveTrain provides an interface to Matlab[55] for post analysis, supporting interactive browsing and random-access loading.

WaveTrain is layered atop tempus[56], a general purpose simulation tool developed by MZA, drawing heavily upon a number of legacy simulation codes as well as codes such as the Atmospheric Compensation Simulation[57], ACS, and Yet Another Propagation Simulation[58], YAPS.

This software was determined to be one of the two acceptable software models for this project. While there is a fee for WaveTrain, is it available free-of-charge to contractors and government personnel working on U.S. government projects. Because of this, WaveTrain was acquired in house as a backup. Besides the low cost (free), this software is also capable of fast Fourier transform (FFT) beam propagation method and it allows for customized phase screens, such as that determined in chapter 4. The only drawbacks to this model are the extensive learning curve required to use the base model, tempus, and that the long propagation distances for which the model was designed. For relatively short propagation distances i.e. less than 10 meters, as required for this project, may pose some problems for the simulation

5.6 LightPipes

OKO Technologies[59] beam propagation toolbox, LightPipes[60], originally designed for UNIX or DOS, now ported over to work with Mathcad and Matlab, is designed to model coherent optical devices. Written in C, this toolbox consists of a number of functions, each representing an optical element or a step in the light propagation. There are a variety of apertures, intensity filters, beam-splitters, lenses and models of free space diffraction in the toolbox. There are also more advanced tools for manipulating the phase and amplitude of the light. The program operates on a large data structure, containing square two-dimensional arrays of complex amplitudes of the optical field of the propagating light beam.

While not a stand alone program, LightPipes is designed to work as a module within Matlab. Advantages of this are: the graphics, animation, and other features of Matlab can be combined with the LightPipes commands, the use of variable arguments in

a function calls and handles complex data structures in a very simple way, enhanced flexibility, and fast execution.

LightPipes can be used in part or in whole, depending on how complex the optical system is. Because it relies on an outside programming environment, the learning curve for LightPipes is relatively small. With each command having its own help screen and listed inputs and outputs, the user can readily pick and choose the proper commands for an optical element. For the purpose of the work being accomplished in this research, LightPipes was found to be very suitable for user-defined phase screens and for beam manipulation, and the code was easy to use and understand. Also, LightPipes contains several methods of beam propagation (described below). The cost was only $399. As a result, this toolbox was chosen to be the primary software for this project.

5.6.1 LightPipes Beam Propagation - Direct Integration

Of the key aspects being looked at for the accomplishment of this work, the manipulation of phase and the type of beam propagation model were the two most prominent. While phase manipulation in LightPipes is rather trivial, beam propagation is not. Because of this, the method of beam propagation will now be discussed. LightPipes contains four beam propagation methods: direct integration, direct integration as a convolution using a fast Fourier transform approach, the fast Fourier transform propagation, and a finite difference method.

The first method is direct integration. Recall from chapter 3 that the solution to the wave equation can be found using a variety of methods. The method implemented in

the LightPipes direct integration propagation is the Fresnel-Kirchoff diffraction integral. Ultimately this integral takes the form of equation 5.1[61] of:

$$E(x_1, y_1, z) = \frac{k}{2\pi i z} \iint E(x, y, 0) e^{\left(ik \frac{(x-x_1)^2 + (y-y_1)^2}{2z} \right)} dx dy , \quad (5.1)$$

where the functions $E(x_1, y_1, z)$ and $E(x, y, z)$ are defined on rectangular grids. The number of operations in the direct calculation of the Fresnel-Kirchoff integral required increases as the grid size raised to the 4th power. Since it does not suffer from reflective boundaries, the cross section of the field distribution can be matched with the computational boundary. The advantage of this scheme is that it can have an arbitrary sampling size and a square grid at the input plane. The propagation can also take place between differently sized grids. For example, the grid size can be adapted as the beam expands. However, this method requires a lot of computer memory and computation cycles. For example, a 64x64 grid size requires about 16 million operations. This will prove to be limiting factor when the different methods of propagation are compared in the results.

5.6.2 LightPipes Beam Propagation – Fast Fourier Transform

A second method of propagation in LightPipes is using the fast Fourier transform, FFT. Recall from chapter 3 the Fourier transform in equation 3.28. This equation is again represented here in equation 5.2.

$$F(\alpha,\beta,z) = \iint g(x,y,z) e^{[-ik(\alpha x+\beta y)]} dxdy \qquad (5.2)$$

If one considers the wave function U in two planes: $U(x,y,0)$ and $U(x,y,z)$, where $U(x,y,z)$ is the resultant is of the propagation of $U(x,y,0)$ to the distance z, then equation 5.2, gives the Fourier transforms of $F(\alpha,\beta,0)$ and $F(\alpha,\beta,z)$, as seen in equations 5.3 and 5.4.

$$F(\alpha,\beta,0) = \iint U(x,y,0) e^{[-ik(\alpha x+\beta y)]} dxdy \qquad (5.3)$$

$$F(\alpha,\beta,z) = \iint U(x,y,z) e^{[-ik(\alpha x+\beta y)]} dxdy \qquad (5.4)$$

In the Fresnel approximation, the Fourier transform of the diffracted wave function is related to the Fourier transform of the initial function via the frequency transfer characteristic of the free space $H(\alpha,\beta,z)$, given by equation 5.3.[62]

$$H(\alpha,\beta,z) = \frac{F(\alpha,\beta,z)}{F(\alpha,\beta,0)} = e^{[-ikz(1-\alpha^2-\beta^2)^{1/2}]} \qquad (5.5)$$

These equations provide a symmetrical relationship between the initial and diffracted wave functions in the Fresnel approximation. If applied in the order of equation 5.3, 5.5, and then 5.4, they result in the diffracted wave function. Applied in the reverse order of equation 5.4, 5.5, and then 5.3, they allow for reconstruction of the initial wave function from the result of diffraction.

While the details of FFT are discussed elsewhere[63], a brief overview is given here. The FFT is basically a modification of the discrete Fourier transform, DFT, algorithm.

The DFT algorithm takes the continuous Fourier transform of equation 5.2 and breaks it up in sections based on letting the function $U(x,y)$ become $U(x_m,y_m)$ where m=0...N-1, giving the DFT as equation 5.6.

$$F(\alpha_n, \beta_n, z) = \sum_{\alpha_n=0}^{N-1} \sum_{\beta_n=0}^{N-1} g(x, y, z) e^{[\frac{-ik(\alpha_n x + \beta_n y)}{N}]} \qquad (5.6)$$

The FFT reduces the number of computations needed for N points from $2N^2$ to $2Nlg(N)$, where lg is the base-2 logarithm. Applying equation 5.3, 5.5, and then 5.4, can be implemented numerically using FFT on a finite rectangular grid with periodic boundary conditions. It results in a model of beam propagation inside a rectangular wave guide with reflective walls. To approximate free-space propagation, wide empty guard bands have to be formed around the wave function defined on a grid. To eliminate the influence of the finite rectangular data window, Gaussian amplitude windowing in the frequency domain could be applied to reduce these effects to allow for extensive analysis of these computational aspects.

5.6.3 LightPipes Beam Propagation – Direct Integration as Convolution

A combination of the direct integration as well as using the FFT can be applied. This approach is numerically implemented with direct summation using the FFT method described in section 5.6.2 and applying it to the Fresnel-Kirchoff diffraction integral in equation 5.1. This integral may be converted into a convolution form which can be efficiently computed using FFT. This method is free from the many drawbacks of the spectral method given by the sequence of the algorithm of implementing equation 5.3,

5.5, and 5.4, yet is it still very fast due to its use of FFT for computing of the integral sums. The summation of the Fresnel-Kirchoff diffraction integral is accomplished by using the trapezoidal rule. While it is slightly slower than the FFT approach, about 2 to 5 times, and uses approximately 8 times more memory than the FFT approach,[62] it does however allow for "more honest" calculations of near and far-field diffraction. Since it does not require any protection bands at the edges of the region, the model may be built in a smaller grid, therefore the resources consumed and time of execution are comparable or even better than that of FFT propagation approach.

5.6.4 LightPipes Beam Propagation – Finite Difference Method

The final method of propagation in the LightPipes toolbox is finite differences. It can be shown that the propagation of the field U in a medium with complex refractive coefficient A, is described by the differential equation given in equation 5.7.[62]

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + 2ik\frac{\partial U}{\partial z} + A(x, y, z)U = 0 \qquad (5.6)$$

To solve this equation, one can re-write it as a system of finite difference equations:

$$\frac{\partial^2 U}{\partial x^2} = \frac{U_{i+1,j}^{k+1} - 2U_{i,j}^{k+1} + U_{i-1,j}^{k+1}}{\Delta x^2} \qquad (5.7)$$

$$\frac{\partial^2 U}{\partial y^2} = \frac{U_{i,j+1}^{k} - 2U_{i,j}^{k} + U_{i,j-1}^{k}}{\Delta y^2} \qquad (5.8)$$

$$2ik\frac{\partial U}{\partial z} = 2ik\frac{U_{i,j}^{k+1} - 2U_{i,j}^{k}}{\Delta z} \tag{5.9}$$

$$A(x,y,z)U = A_{i,j}^{k+1}U_{i,j}^{k+1} \tag{5.10}$$

Next, collecting like terms results in a 5-diagonal and is solved by the standard elimination method, also known as the double sweep method.[64] This scheme is absolutely stable, meaning that this variant is explicit with respect to the index i and implicit with respect to the index j. One step of propagation is divided into two sub-steps: the first sub-step applies the described procedure to all rows of the matrix, the second sub-step changes the direction of elimination and the procedure is applied to all columns of the matrix.

The tri-diagonal system of linear equations has a solution of which describes the complex amplitude of the optical field in the layer $z+\Delta z$ as a function of the field defined in the layer z. If one equates $\Delta x = \Delta y = \Delta$, and uses coefficients of A_i, B_i, C_i, and F_i as denoted by equations 5.11-14,

$$A_i = -\frac{1}{\Delta^2} \tag{5.11}$$

$$B_i = -\frac{1}{\Delta^2} \tag{5.12}$$

$$C_i = A_{i,j}^{k+1} - \frac{2}{\Delta^2} + \frac{2ik}{\Delta z} \tag{5.13}$$

$$Fi = \frac{2ik}{\Delta z}U_{i,j}^{k} - \frac{U_{i,j+1}^{k} - 2U_{i,j}^{k} + U_{i,j-1}^{k}}{\Delta^2} \tag{5.14}$$

Then the final solution looks like equation 5.15.[62]

$$-A_i U_{i-1,j}^{k+1} + C_i U_{i,j}^{k+1} - B_i U_{i+1,j}^{k+1} = F_i \qquad (5.15)$$

The main advantage of this approach is the ability to take into account nonuniform absorption and refractive indices. For example, a waveguide with complex three-dimensional distribution of refraction index and absorption coefficient, where both are defined as real and imaginary components of the three-dimensional matrix A_{kji}, can be simulated easily.

As the scheme is absolutely stable, there is no stability limitation on the step size in the z direction. However, large steps can cause high-frequency errors. Therefore the number of steps should be determined by trial and error, especially for strong refractive index and absorption coefficient variations. There are several methods of handing the boundary conditions. Setting the boundary elements to zero amplitude is the simplest method. However, this creates nonphysical reflections at the grid boundaries. The influence of these reflections in many cases can be reduced by introducing an additional absorbing layer in the proximity of the boundary, with the absorption smoothly increasing towards the boundary to reduce the reflection at the absorption gradient.

CHAPTER 6

ANALYSIS AND RESULTS

6.1 Verification of Code

To begin the analysis, the performance of the code needs to be verified to show that it performs as expected. To accomplish this, a series of tests were performed of which the output or parameter being verified could be compared with an expected or known output. This verification process occurred in a series of steps. The first step involved making initial checks to verify that the code was properly reading in the supplied information. This entailed a series of visual checks of the read-in images, beam parameters, and phase information.

The second step was to verify that the information being used inside the model was correct. For example, the proper values for the phase being used in the images, the phase screens' accuracy, the setup of the laser beam, and the incorporation of the phase screen were all checked. This step served as the internal verification that the correct parameters were being used in the proper methods so that the information being used in the calculations was accurate.

Next, the calculations of the simulation were checked. Most of the checking calculations were performed by hand or with separate coding to double check that the formulas presented in the model were correct. Calculations too long to do by hand, such

as the propagation, were checked against final output results that could be calculated. Issues that were dealt with in this part of the verification were; whether the phase of the laser beam matched the expected phase, if the output at different propagated distances matched the calculated output at these same distances, and whether or not the different methods of propagation agreed.

Finally, a verification that the code reacted to changes being pursued in the model needed to be accomplished. These changes included using the different methods of propagation mentioned in chapter 5, being able to vary the wavelength of the laser being used, shifting the laser beam around in the given space, and adding rotation to the phase screens to simulate the beam coming in at a different incident angles.

The results of the verification process are provided in this chapter using several options. First, the verification of items to check is presented in table form. This was accomplished by creating a list of the parameters to check and then going through and verifying them by one of three options. These options include verification by inspection, verification by trial, and verification by calculation. The verification by inspection is the process of going in to the code and outputting the parameter to be verified by posting it to the output screen, displaying it as an image, displaying it as a plot, or by writing in flags to verify that parameters are met. The second option for verification is that of verification by trial. This occurs by letting the code perform the calculations and then check the output, either by inspection or comparing against calculations. This leads to the third aspect of verification, verification by calculation. Here, predicted outputs can be figured out by other methods, i.e. by hand calculations or use of other software, and

compared to the calculations performed by the simulation. The results of the parameters for verification are presented in tabular form in table 6.1.

Table 6.1 Code Verification Processes

Items to Verify	Verified By		
	Inspection	Trial	Calculation
Set selection	✓		
Run selection	✓		
Correct image selected	✓	✓	
Boundary layer determination	✓	✓	
Layers filled correctly	✓	✓	
Image rotation	✓	✓	✓
Refractive index determination	✓	✓	✓
Mixing layer determination	✓	✓	
Refractive index insertion	✓	✓	
Change of index option	✓	✓	
Units determination	✓	✓	
Pixel height	✓		✓
Wavelength designation	✓	✓	
Optical path length calculation	✓	✓	✓
Phase determination	✓	✓	✓
Plane wave profile	✓	✓	
Gaussian beam profile	✓	✓	
Beam width	✓	✓	
Beam center	✓	✓	
Beam intensity	✓	✓	
Initial phase	✓	✓	
Phase screen size	✓	✓	✓
Phase insertion	✓	✓	
FFT propagation	✓	✓	✓
Direct integration propagation	✓	✓	✓
Integration using convolution propagation	✓	✓	✓
Finite difference propagation	✓	✓	✓
Beam intensity after propagation	✓	✓	
Correct measured phase selection	✓	✓	
Phase correlation	✓	✓	✓
Beam shifting	✓	✓	✓
Beam divergence calculation	✓	✓	✓

To obtain the verification of the calculations mentioned in table 6.1, a cube was designed with two separate refractive indices, n_1 and n_2, split down the center at a 45 degree angle. This cube can be seen in figure 6.1.

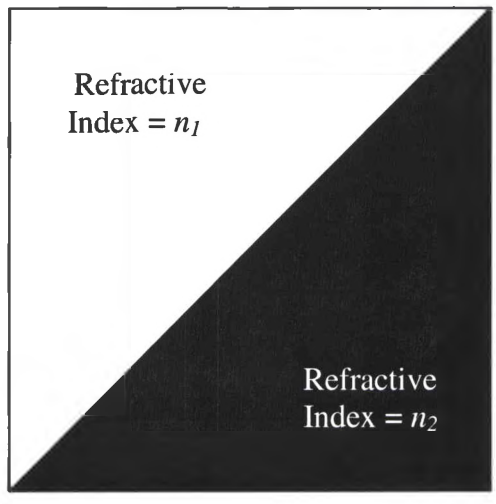

Figure 6.1. Two Tone Refractive Index Cube

The reasoning behind using the figure 6.1 design of the cube is that many different aspects exist that can be easily checked. First off, the phase, as suspected based on the shape of the cube, should be linear. As a result, a visual check is sufficient to see that it is being substituted in correctly. Also, while the visual inspections are readily apparent, and the calculation of phase is predictable, the actual "plugging in" of the phase can be double checked using known equations. The first equation that can be used is that of Law of Refraction, or Snell's Law, given in equation 6.1.[65]

$$n_i \sin\theta_i = n_t \sin\theta_t \qquad (6.1)$$

By knowing that the cube is split down the diagonal, giving a 45° slant, the angle of incidence can be calculated. Coincide this with the knowledge of the refractive indices of the cube, and the final output angle can be computed. A diagram is given in figure 6.2 for the calculation of the final output angle.

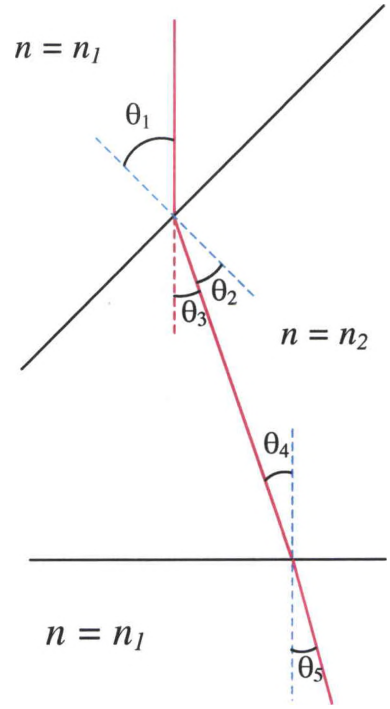

Figure 6.2. Beam Geometry for Angle Calculation

Using the geometry of figure 6.2 and equation 6.1, the final output angle is determined to be equation 6.2.

$$\theta_5 = \sin^{-1}\left(\frac{n_2 \sin\left(\theta_1 - \sin^{-1}\left(\frac{n_1 \sin(\theta_1)}{n_2}\right)\right)}{n_1}\right) \qquad (6.2)$$

The accuracy of the model could then be checked by propagating a laser beam out to a known distance after passing through the cube, measuring how far the beam traveled off center, and doing some trigonometry to find the transmitted angle. This was accomplished by designing a test loop to evaluate this part of the code.

The test loop performed as follows. First, the refractive index, n_1, was chosen and held at a constant. The opposing refractive index, n_2, was then varied in increments, increasing the difference between n_1 and n_2. The resulting output beam was then compared with an undisturbed beam to calculate the angle of refraction. Then, using equation 6.2, the result was compared with the calculated angle, and a percent deviation was calculated. The percent deviation calculation was most appropriate here in that there is a true angle that could be calculated that the simulation was trying to match. The formula for the percent deviation is given by equation 6.3 of:

$$\%Dev = \frac{(X_m - X_t)}{X_t} \times 100\%, \qquad (6.3)$$

where X_m is the measured value

X_t is the true, or calculated value.

The percent deviation was calculated to help show at what locations the simulation began to fail, or in the least, give an idea of the accuracy of simulation in the given realm being explored. This percent deviation was also the basis for comparison of the different methods of propagation.

6.2 Comparison of Propagation Methods

The comparison of simulated verses calculated output angle, using the previously mentioned cube, was accomplished coinciding with the verification of other parameters. Recall that the different methods of propagation, with accompanying changes of refractive index, also need to be verified. Here, by combining these different parameters into another series of tests, this was completed.

6.2.1 Percent Deviation of FFT Method

The first method of propagation tested was the use of the fast Fourier transform method, described in depth in section 5.6.2. This propagation, denoted in LightPipes as LPForvard, was the fastest of the four methods presented. The method was verified using the following process. A test was designed to replicate the recording experiment with the exception of the use of the two-tone cube described in section 6.1 in place of the jet stream images. The refractive indices used in the test were as follows. The first refractive index, n_1, was held constant at a value of 1.000. The second refractive index, n_2, varied in value from 1.000 to 1.003. While the difference in change of index may seem small here, note that the indices of refraction determined to be occurring in the recording experiment were 1.00023 to 1.00031, so the results were actually tested out to a factor of ten larger.

By using equation 6.2 and knowing values of refractive index, the expected output angle was calculated. This angle was then compared to the simulation output angle. The simulation output angle was found by propagating the beam out a known distance, finding the center of the beam, using a Gaussian curve fit, finding the shift from the

original position, and then using the inverse tangent to find the angle. The two angles were then put into equation 6.3 to find the percent difference. The results of propagation using the FFT method can be seen in figure 6.3.

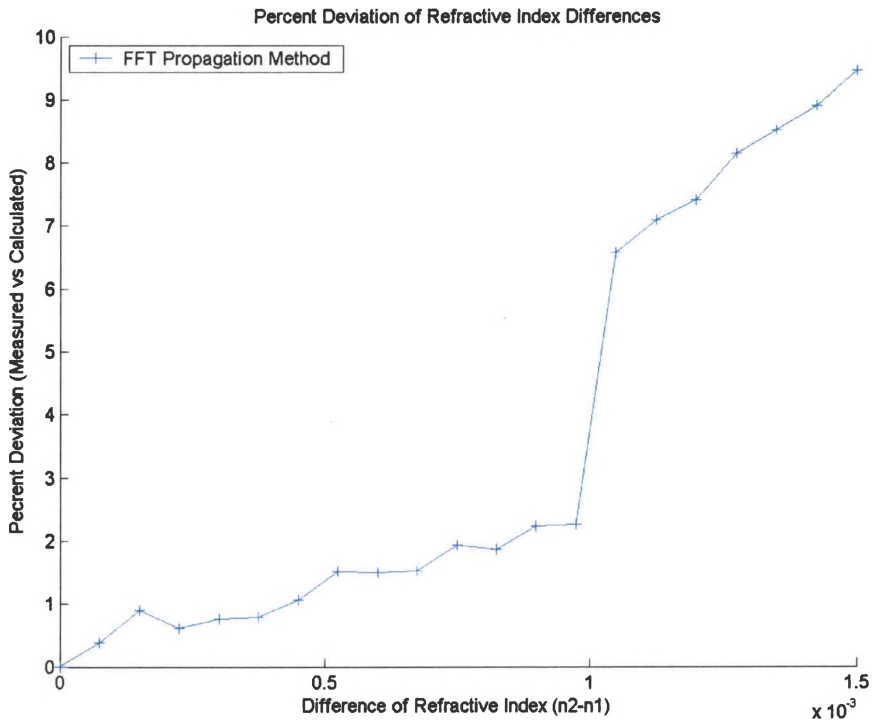

Figure 6.3 Percent Deviation Using FFT Method

Of the issues to keep in mind during the use of FFT propagation method was that, recall from section 5.6.2, the algorithm is basically a model of beam propagation inside a square wave guide with reflecting walls positioned at the grid edges. If the intensity near the walls is not negligible small, the beam begins to reflect back on itself, resulting in unphysical reactions performed by the algorithm. As a consequence of this, care must be taken in propagating the plane wave to a distance comparable with the far field, i.e. D^2/λ where D is the diameter of the beam, and λ is the wavelength. To propagate the

beam to this distance the size of the grid must be larger than the beam itself. Thus the grid edges must be far enough away from the propagating beam. During the propagation accomplished to produce figure 6.3, due to the small change in refractive index and fact that the distance propagated was only one meter, the grid edges did not play a role in effecting the propagation. However, because the cube did shift the beam, in the ladder parts of the verification, when the refractive index was 0.002 or higher, the output beam was visually inspected to verify that these edge effects were not a factor.

6.2.2 Percent Deviation of Convolution Method

The second method of propagation tested was that of direct integration as a convolution using an FFT approach, discussed in section 5.6.3. This propagation, denoted in LightPipes as LPFresnel, making reference to the Fresnel-Kirchoff diffraction integral, while being the second fastest performing propagation method was also the second most accurate. This method was verified using the same process as the FFT approach using the same code and changing only the propagation from LPForvard to LPFresnel. The resulting output showed similar to slightly better performance over the FFT propagation, however, the time it took run the simulation a little more than doubled. The total run time to obtain the 21 point diagram shown for the percent deviation of the direct integration using convolution seen in figure 6.4 took approximately 35 minutes to accomplish. The FFT 21 point run of figure 6.3 was completed in 15 minutes. Both the accuracy and the run time were taken into account in the determination of the method of propagation for use in the simulation of the recording experiment.

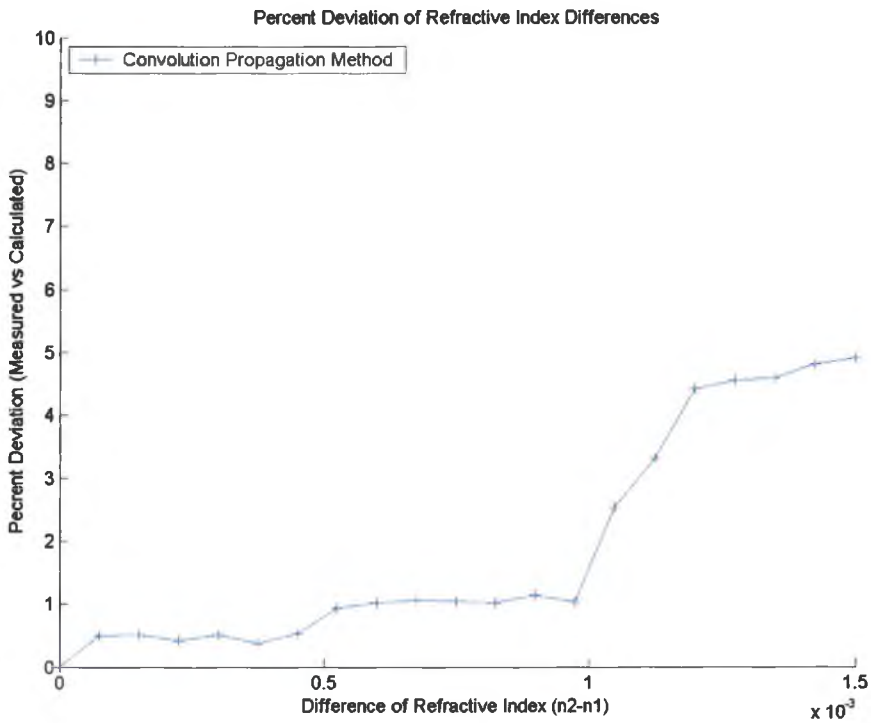

Figure 6.4 Percent Deviation Using Convolution Method

As discussed in section 5.6.3, this method is free from many drawbacks of the FFT method due to the sequence that the algorithm implements. This method of propagation does not require any protection bands at the edges of the region. As a result, the model may be built in a smaller grid, thereby conserving the resources consumed and time of execution. In the code in which it was implemented here, a larger grid was used to better match up with the recording experiment. The code could have been altered to allow for a smaller grid size which may have resulted in processing time performance comparable or even better than that of FFT propagation.

6.2.3 Percent Deviation of Finite Difference Method

The third method of propagation tested was that of the finite difference method, described in section 5.6.4. Denoted as LPSteps in the LightPipes code, this method is slightly more work to implement into the experiment. While the FFT and convolution methods are easily swapped by changing just a word in the code from LPForvard to LPFresnel, implementing LPSteps requires an additional line or two. This requirement comes from the possibility of being able to take into account a medium that is of uniform diffraction, absorption, and refraction. While this method works much faster than the FFT, convolution, and direct integration methods for *one* step of propagation, the distances for which the propagation in the one step is valid becomes an issue. Large steps in the finite difference method propagation cause high-frequency errors. Therefore, the number of steps needed is determined by trial. It was recommended from the LightPipes manual that in order to obtain a good result at a considerable distance, many steps should be accomplished. After a series of trials, it was determined that fifty steps gave an approximate correlation with the FFT and convolution methods. However, the time required to perform this operation with this number of steps increased five fold over the FFT method.

Of issues to overcome with this method is the fact that, similar to the FFT method, zero amplitude boundary conditions are used for this method. This again creates the problem of the wave reflection at the grid boundary. To aid in the reduction of this effect, the designers of LightPipes have built in an absorption layer along the grid boundaries occupying 10% of grid from each side to help prevent reflections. It was suggested, however, that an additional absorbing layer may be used in the in the

proximity of the boundary, with the absorption smoothly increasing towards the boundary to help reduce this effect coinciding with the use of many steps.[62] The percent deviation of this method per change in refractive index is given in figure 6.5.

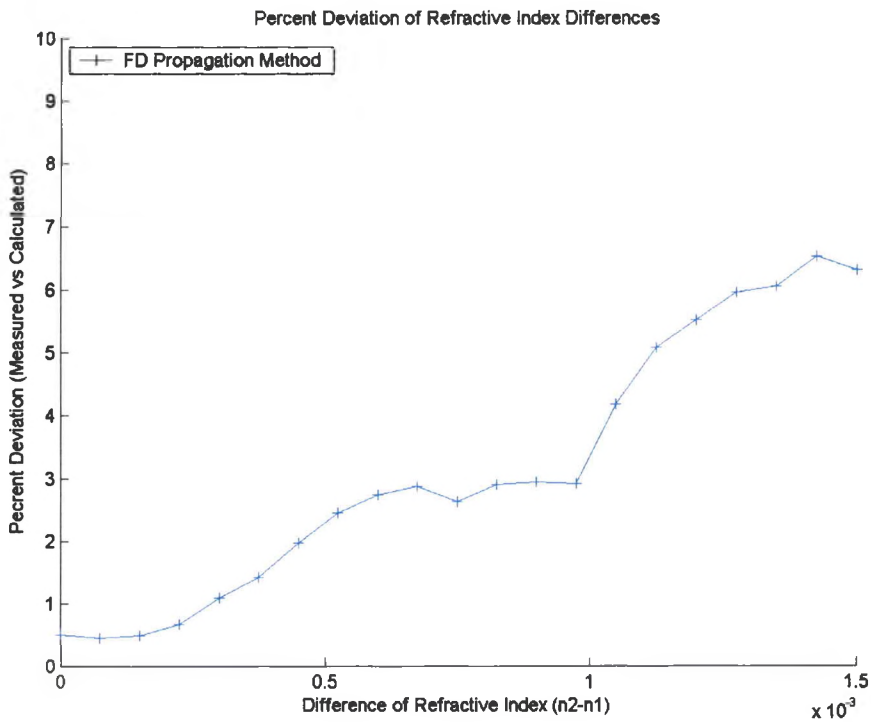

Figure 6.5 Percent Deviation Using Finite Difference Method

6.2.4 Percent Deviation of Direct Integration Method

The fourth method of propagation given in LightPipes is that of direct calculation of the integral as described in section 5.6.1. While this method is proclaimed to be the most accurate method, it also requires the most resources. Recall from the previous section that the number of operations required to perform the propagation follows as N^4, where N is the size of the grid to be used. In the experiment performed, the grid size was 1024 x 1024. Consequently, the number of calculations required to perform this

propagation is 1024^4, far too many operations to be performed with the given resources of the computers being used to run this simulation. As a result, it was determined that this method would be eliminated from testing procedures.

6.2.5 Choice of Propagation Method

To determine the propagation method to be used in the simulation, several factors needed to be considered. First, the code must have been easily implemented. Though a slight amount of work was required to implement the finite difference method, all of methods were, for the most part, easily implemented.

The second factor to consider was that of accuracy. While the most accurate method, direct integration, was eliminated because of processing time, the other methods proved to very similar in accuracy in the ranges of which the simulation was to occur. Figure 6.6, shown on the following page, gives a side-by-side comparison of the FFT, FD, and convolution propagation methods when propagated out to two meters. The convolution method gives less that one percent deviation out to a refractive index difference of 0.001. While this indeed was the most accurate of the three, the other methods are nearly as accurate with the FFT method giving just under two percent and the FD method giving less than three percent difference for the same range.

While the percent difference does vary a couple of percent, when the range approaches that of the one determined to be occurring in the recording experiment, i.e., down in the .0007 range, all three methods show approximately a .5% deviation. As a result, all of the methods were deemed acceptable in the performance of accuracy.

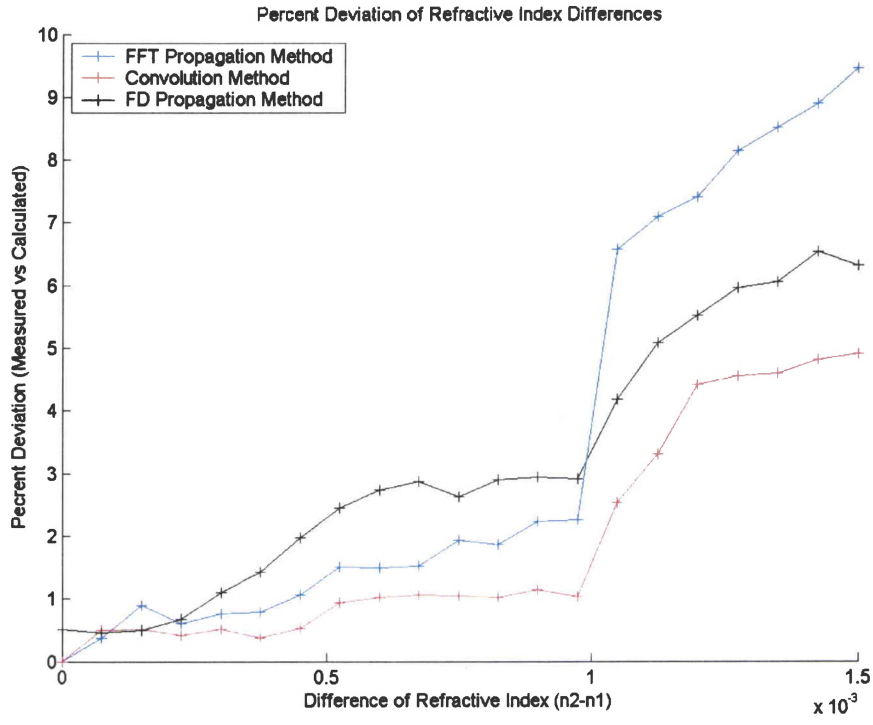

Figure 6.6 Percent Deviation Method Comparison

The third deciding factor for the final propagation method is that of the processing time. While the time needed to perform the operations will vary depending on the processing equipment, the computer being used to run the simulation was a Dell LATTITUDE D800 with a 1600 MHz processor with 523,496 KB RAM. To make a far comparison, all of the propagation methods were run on the same machine and the following times were recorded to accomplish the 21 point graphs seen in figures 6.2 through 6.6. The FFT propagation method, denoted as LPForvard, processing time took approximately 15 minutes to accomplish the 21 point graph. The convolution propagation method using an FFT approach, denoted as LPFresnel, took approximately 35 minutes to process the same information. The finite difference propagation method, denoted as LPSteps, recall using a fifty step propagation, took approximately 75 minutes.

The final propagation method, direct integration, denoted as LPForward, ran longer than 2 hours before the runs were terminated due to the extensive amount of time that it was taking to perform the analysis.

As a result of all three of the above factors, the ease of use in the code, the accuracy of the output, and processing time, it was determined that the best approach for use in the propagation of the beam was to use the FFT method, LPForvard, because of its sufficient accuracy, ease of use, and short processing times.

It is of significance to note from figure 6.6 that the maximum difference of change in refractive index was 0.0015, resulting in a calculated output angle of 0.082 degrees using the "two tone" cube. Also note that at refractive index change of 0.001, all of the methods of propagation begin an upward trend increasing the percent difference. After reviewing this further, it was found that at this point, the change in phase over four consecutive rows is greater than 2π. Basically what is occurring is that the sampling rate is becoming an issue, i.e. that the amount of phase change of the medium is too drastic to be properly represented by the single phase screen method. If the experiment was occurring beyond this realm, a two or more phase screen method of propagation simulation would be needed. However, as stated earlier, the experiment was accomplished in the regime of difference of refractive index of 0.0007, so the single phase screen method of beam propagation simulation remains valid.

6.3 Validation of Rotation

Due to the fact that a portion of the investigation of this research includes rotating the input images to determine if the output characteristics of the beam improve or

degrade, the use of rotation in the code also needs to be verified. As a result, the two-tone test block, seen in figure 6.1, was used again, but incorporated into the code for this validation was Matlab's function for rotation of an image, "imrotate". To simulate possibly accounts of jitter or alignment issues in the exposures from the recording experiment, a rotation of ± 2 degrees in the x-z plane was determined to be sufficient. As a result, the known cube was rotated + 2 degrees in the x-z plane (clockwise rotation) and – 2 degrees in the x-z plane (counterclockwise). Figure 6.7 shows the results for the rotation while figure 6.8 and 6.9 show the percent deviation for the rotation.

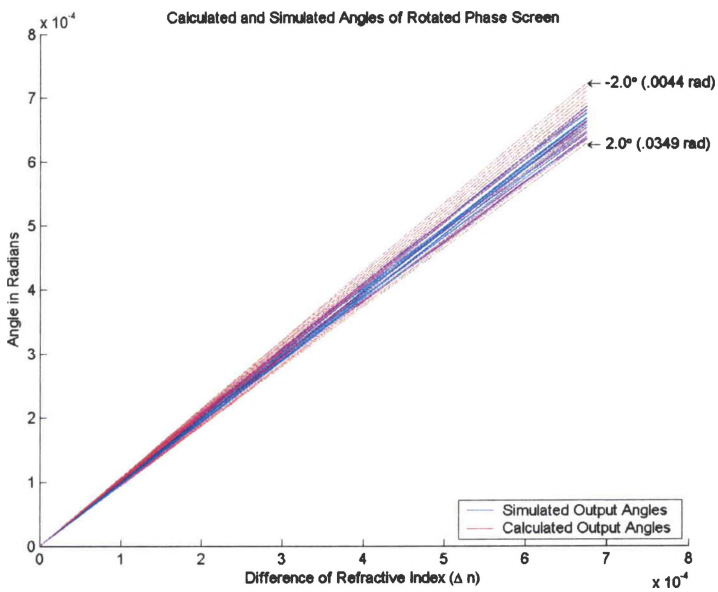

Figure 6.7 Two Degree Rotation Verification

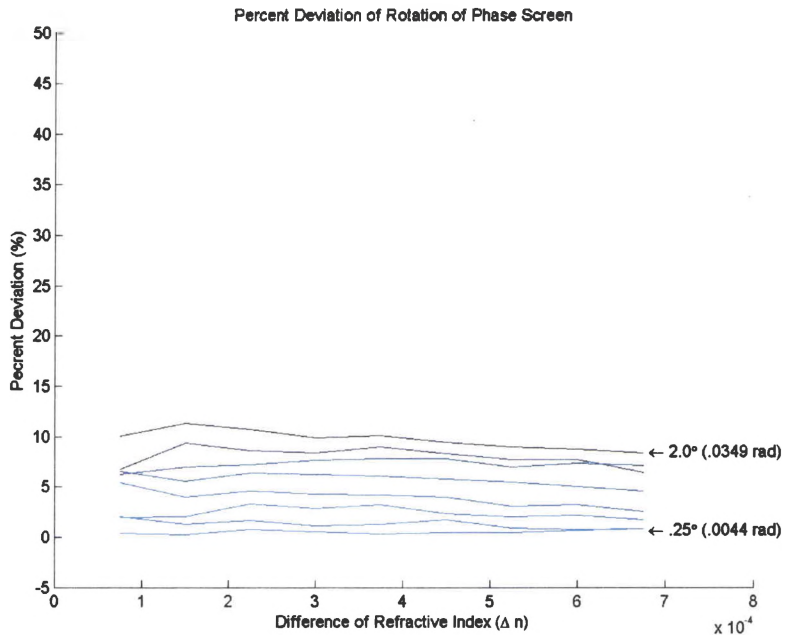

Figure 6.8 Percent Deviation Clockwise Rotation

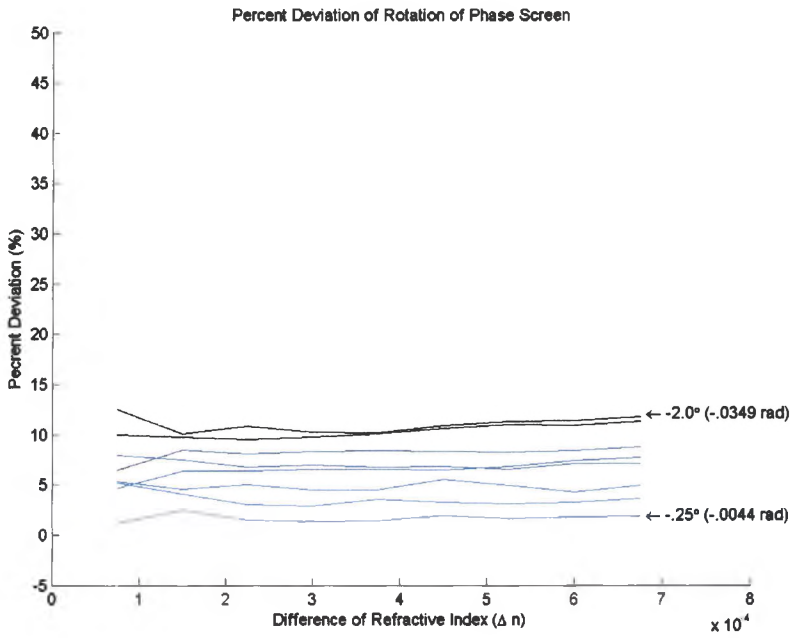

Figure 6.9 Percent Deviation Counterclockwise Rotation

The cause of the rise of the percent deviation in figures 6.8 and 6.9 was a result of the reflective boundaries discussed in section 5.6.2. As the angle increased, the far edge of the beam began to reflect back into itself and caused error in the calculation. To reduce the error, the boundary layer could have been made larger. However, the amount of rotation allowed, ± 2 degrees, was more than substantial to cover any possible errors due to alignment. In fact, ± 1 degree was more sufficient for alignment issues. For this amount of rotation the percent deviation is between zero and one percent, still an acceptable amount of deviation.

6.4 Simulated Output vs. Measured Output

Now that the code has been verified, the next step is the use of the code to predict the outcome of a given situation. To accomplish this, a correlation of the recording experiment is performed. Here, the recorded images of the high speed flow, seen in figure 6.10, are used as the basis for the inputs into the model.

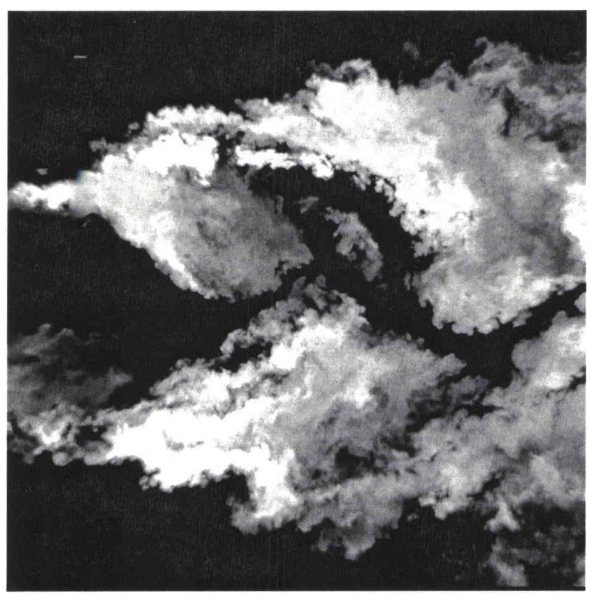

Figure 6.10 Model Input, Flow Image

These images, being two dimensional, are converted into layers of uniform refractive indices, expanded out into a layered block, and then compressed by calculation into a phase screen. This process is described thoroughly in chapter 4. The beam parameters from the recording experiment are then set up in the code and propagated. The actual distance of propagated corresponds to the actual conditions of the recording experiment. The phase screen is then mixed into the beam and the beam is again propagated out to the location of the sensor. This propagation coincides with the distance measured from the jet stream to Shack-Hartmann wavefront sensor, described in chapter 2. After the propagation, the beam characteristics, in particularly the phase of the beam, are calculated. The phase from the simulation is then compared to the simultaneous phase information, an example of which can be seen in figure 6.11, recorded from the experimental conditions.

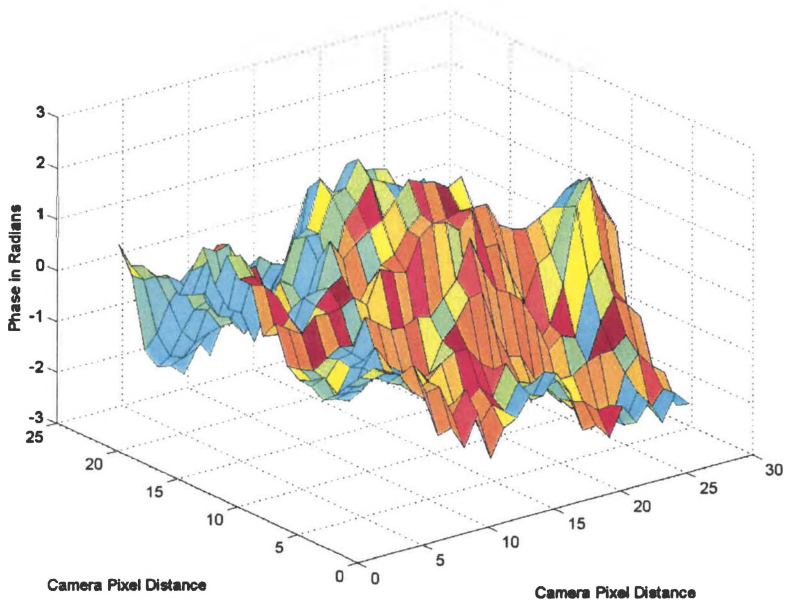

Figure 6.11 Example of Measured Wavefront[17]

The recording experiment produced 129 separate runs grouped together in to five sets. Each of these runs contains an image, similar to figure 6.10, as well as a corresponding wavefront, seen in figure 6.11. To obtain the correlation, a run will be set up for each of the 129 measured runs entailing the same, or as many of the same, parameters that occurred at the recording experiment. This includes using the same laser beam wavelength, the same size beam, and the same propagation distances. The input, as mentioned, will be the image and the final output will be the calculated phase. Note here, however, because only of slice of the three dimensional stream was recorded, the corresponding output phase will only be a slice of the measured phase. As a result, the calculated phase will be correlated with only the corresponding slice of phase.

6.4.1 Output Correlation Results

As mentioned in the previous section, the recording experiment produced five sets of data with varying numbers of runs to each set. Here, the correlation results are broken down by set and then combined for an overall correlation for the entire experiment. However, before the results are listed, the method of correlation needs to be justified. The method of correlation comes in from plugging in the known images and producing an output phase. This calculated phase is then overlapped with the measured phase. During this process, the calculated phase is shifted over a small series of rows to find best fit with the measured phase. This compensates for any beam jitter or small alignment issues obtained during the recoding experiment. Figure 6.12 shows the plane of

correlation and an example of the phase comparison between measured and simulated phases and correlation can be seen in figure 6.13.

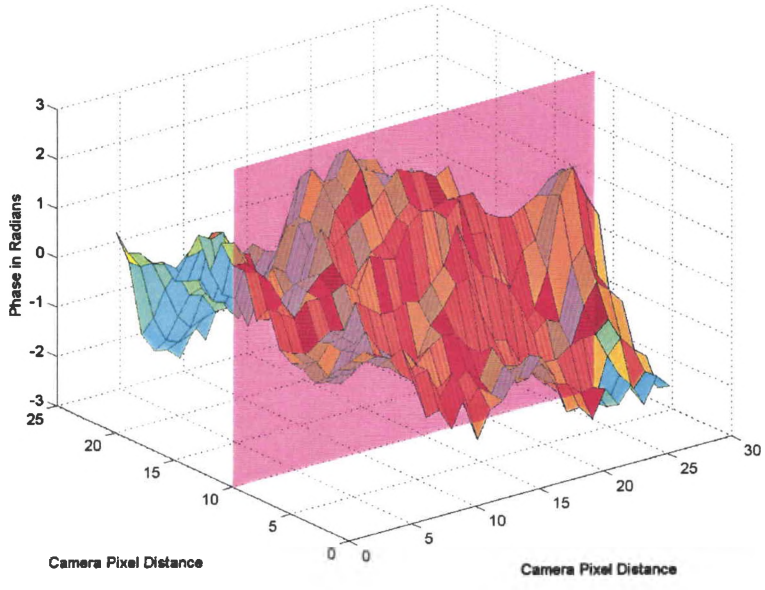

Figure 6.12 Phase Correlation Plane

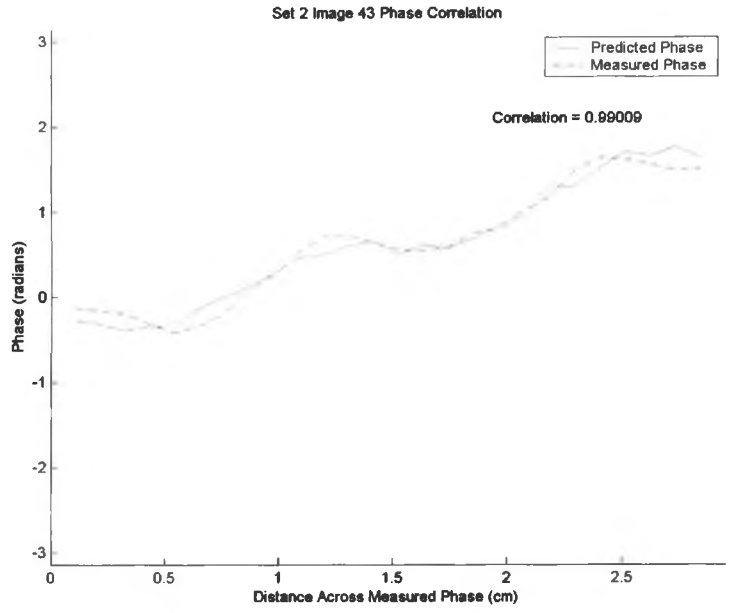

Figure 6.13 Correlation of Predicted and Measured Phase

In figure 6.13, the predicted phase from the model is represented as the solid red line. The measured phase from the recording experiment is represented by the dashed blue line. The x axis gives the distance across the laser while the y axis presents the phase. Note the correlation factor in the figure. This factor is a result of Matlab's correlation function, "xcorr" and is calculated by the following.

In MATLAB, xcorr (x,y) returns the cross-correlation sequence in a length $2*N-1$ vector, where x and y are length N vectors ($N>1$). If x and y are not the same length, the shorter vector is zero-padded to the length of the longer vector. By default, xcorr computes raw correlations with no normalization. The correlation coefficient calculation is given in equation 6.4.[66]

$$\hat{R}_{xy}(m) = \begin{cases} \sum_{n=0}^{N-m-1} x_n + {}_m\, y_n^* & m \geq 0 \\ \hat{R}_{yx}^*(-m) & m < 0 \end{cases} \qquad (6.4)$$

The output vector, if c = xcorr(x,y), has elements given by $c(m) = c_{xy}(m-N)$, m=1, ..., $2N$-1. In general, the correlation function requires normalization to produce an accurate estimate. However, with the correlation performed, the sequence was normalized so the autocorrelations at zero lag are identically 1.0.

The following plots, figures 13-17, show the results of each Set correlation statistics with a average correlation for that particular set, followed by figure 18, showing the results of the entire sets together to obtain an over correlation between the calculated phases and those obtained at the recording experiment.

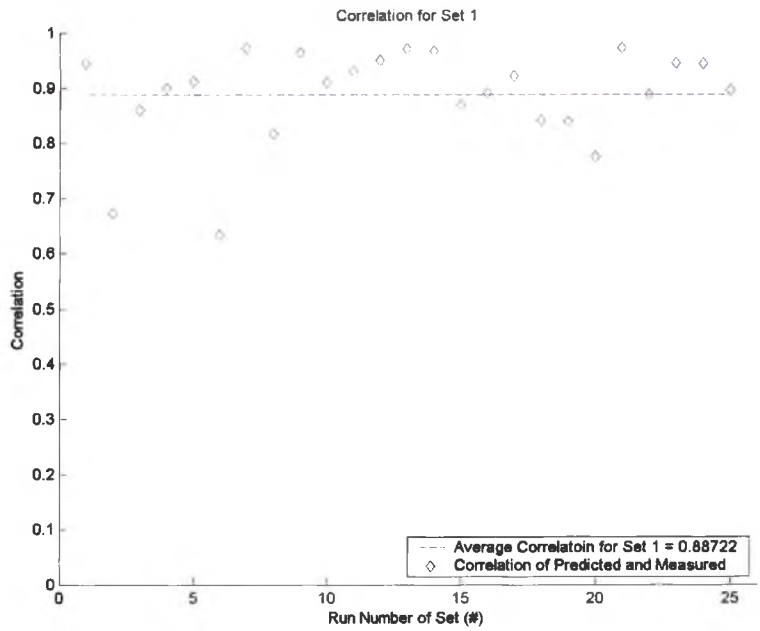

Figure 6.14 Set 1 Correlations

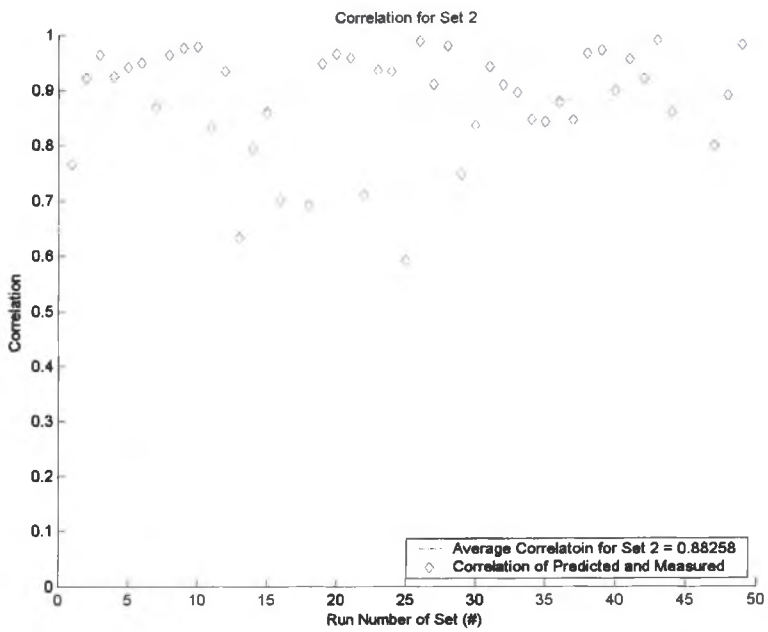

Figure 6.15 Set 2 Correlations

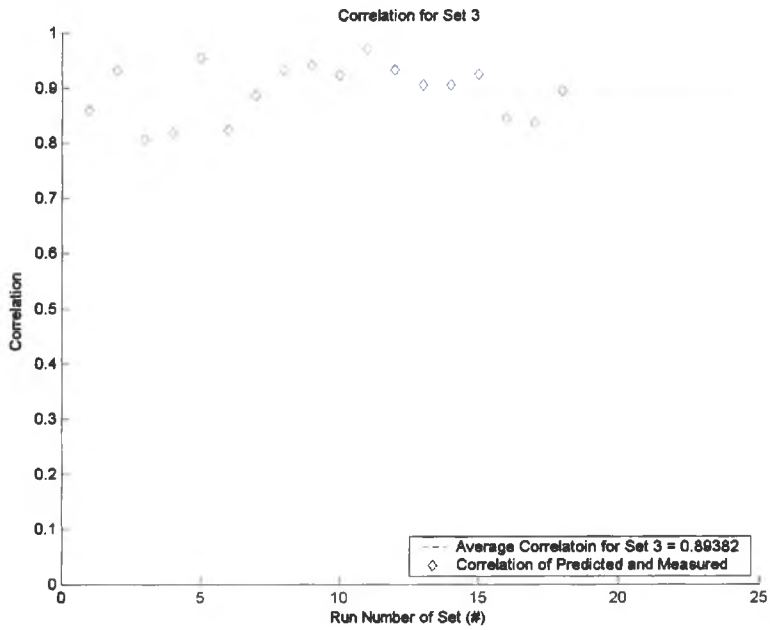

Figure 6.16 Set 3 Correlations

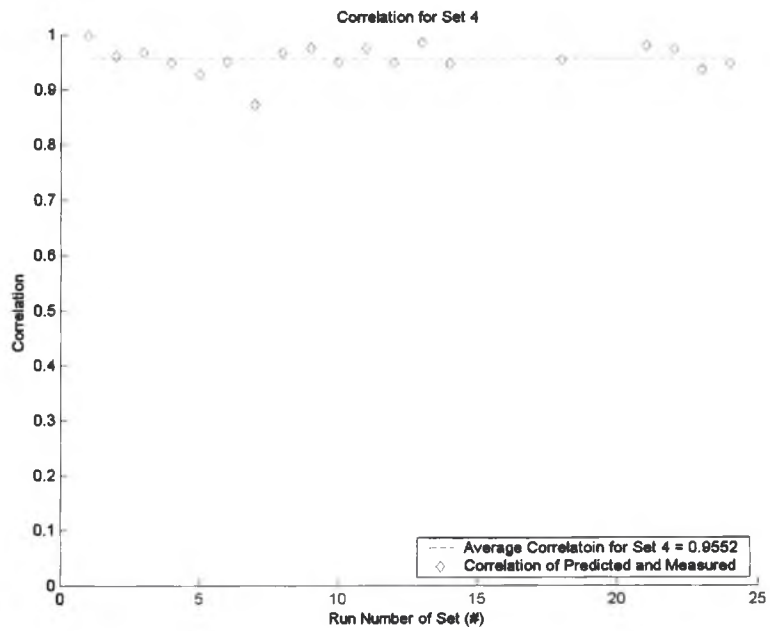

Figure 6.17 Set 4 Correlations

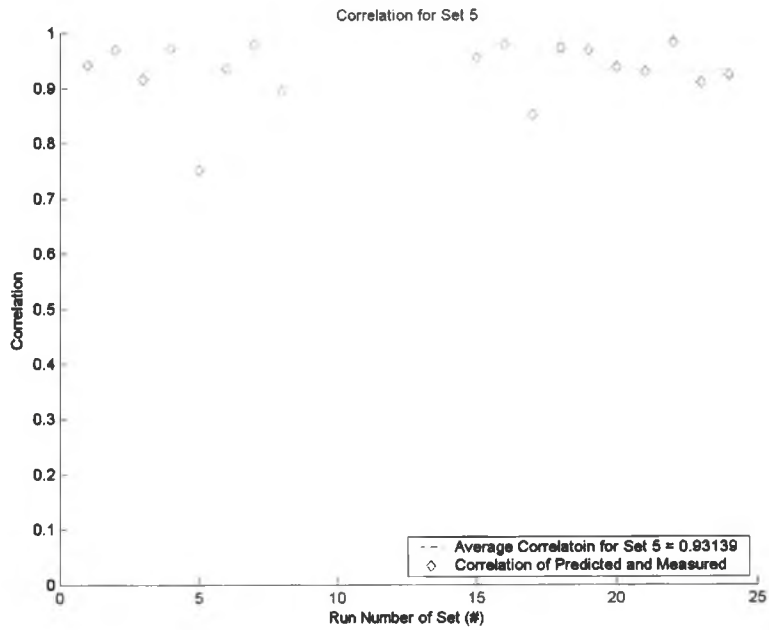

Figure 6.18 Set 5 Correlations

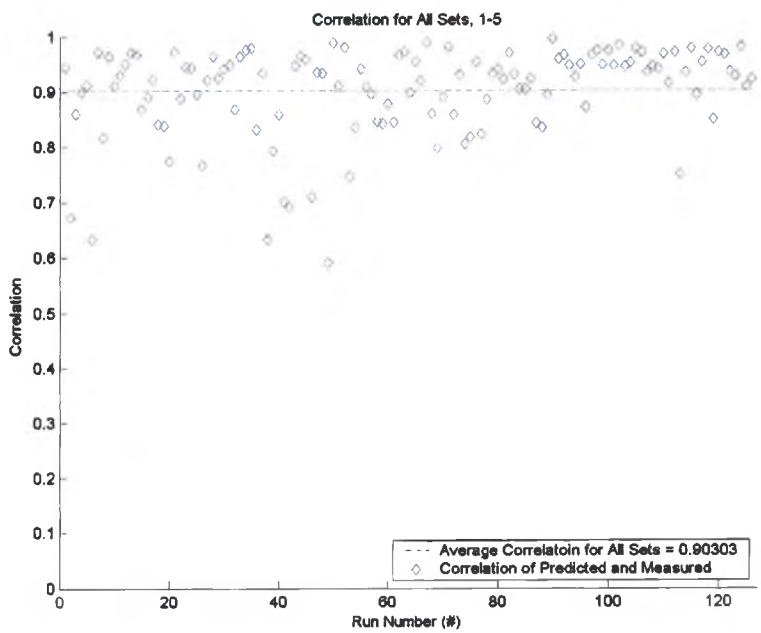

Figure 6.19 Total Correlation for All Sets

As seen in figure 6.19, the average correlation for all five sets is 90.3%. Note of the 129 runs, 85 of the runs had better than 90% correlation. If the acceptable percent correlation was deemed 80%, then 116 runs, or 89.9% of the runs would be deemed accurate.

While the correlations numbers in this research are acceptable and prove that this method of simulation is valid for close approximations over a series of runs, further verification needs to be made to validate its use for the simulation of the recording experiments. Among this validation is taking into account any error caused by shift and rotation and verification that the wavelength, and hence, refractive indices, were simulated properly for the research.

6.4.2 Reducing Error from Shift

While some error may have been introduced into the experiment from a small amount of misalignment or beam jitter, which can be considered a small shift given the distance the beam traveled from the laser to the receiver, this error was taken into account and eliminated in the simulation by finding the "best fit" for the calculated beam over the measured beam by doing a row by row comparison. Any amount of error due to shift, for the most part, was then eliminated in order for correlations to be accomplished in section 6.5.2, so the correlations already have any improvement due to shift. It is of significance to mention here however that of the best fit for the trials, the amount of shift for all cases varied no more than approximately +/- 6 rows of the measured phase in any one direction.

It is also of significance to note here that the row size of the measured phase and the row size of the calculated or simulated phase are not the same. The detector used in the recording experiment was a 1 inch by 1 inch detector and produced an output pattern of 26 by 26 pixels. The simulated phase, however, covering the same one inch square, allowed for 426 by 426 pixel coverage. To correct for this, the simulated phase had to be averaged down to the resolution of the measured phase. A visualization of this can be seen in figure 6.20.

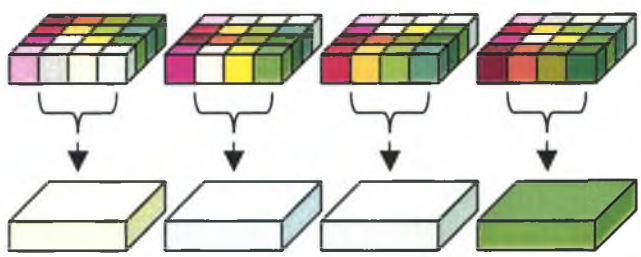

Figure 6.20 Reducing Simulated Phase Resolution by Averaging

In actuality, sixteen pixels of the simulated phase were averaged down to match one pixel of the measured phase by using an averaging function of MATLAB. Now, going back to the statement of shift occurred more than +/- 6 rows of measured phase, which correlates to a simulated phase, before resolution reduction, to nearly +/-98 rows. While this may seem like a large amount, consider that each row had a pixel height of approximately 0.067 millimeters, so the corresponding actual amount of maximum shift that occurred at the camera was only slightly more than half of a centimeter, or +/- 0.658 cm to be exact.

6.4.3 Output Correlation with Wavelength Variance

The next step in validating the model is to verify that the simulated conditions with respect to wavelength, i.e. refractive index which is then used to calculate the phase, matching the actual conditions measured in the recording experiment are in agreement. Because the exact wavelength that was used during recording experiment was known, performing a wavelength variance analysis provides a method of performing a check of the phase calculation accuracy. The results should show the best correlation around the given wavelength of 532 nm and drop off as the wavelength increases or decreases.

To accomplish this, each simulated phase after propagation was compared to the measured phase with a best fit correlation as mentioned in the previous section. The lines and rows for which this correlation occurred were recorded for the run and used in the following. Each run was then put through a loop, varying the laser beam wavelength from 300 nm to 800 nm in increments of 50 nm. A correlation between this new simulated phase for each wavelength and the measured phase was then accomplished. The correlation for each run was then plotted as a function of wavelength. A sample of the correlation as a function wavelength is given in figure 6.21.

To help verify that the simulation being accomplished is accurate, a second type of plot is needed. After all of the correlations of wavelengths for all of the runs were completed, the results of the highest correlation for run as a function wavelength were plotted. Basically, this gives a histogram of the results, showing the number of times a particular wavelength resulted in the highest correlation for a particular run. The histogram plot is shown in figure 6.22.

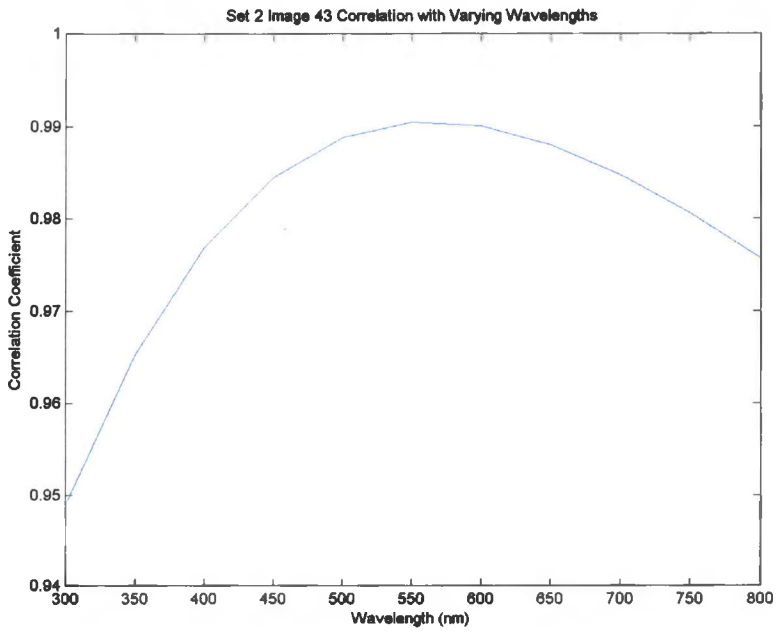

Figure 6.21 Sample Correlation with Varying Wavelengths

In figure 6.21, the correlation shows that the best correlation between the simulated phase and measured phase of set number 2, trial number 43, occurs at 550 nm. Note, the actual wavelength used in the experiment was 532.8 nm. In using increments of 50 nm, 532.8 nm rounds up to the 550 nm wavelength. Note, however, that figure 6.21 is not a total representation of all of the trials. In fact, while several runs proved to be similar nature to figure 6.21, a variety of the runs proved to show better correlation at one of the other wavelengths.

The histogram plot is shown in figure 6.22. Note this plot only represents runs that had a better initial correlation greater than the entire set average of 90.3%. The reasoning behind this is that the analysis of the runs having an initial correlation below the average for the most part showed the correlation improvement going beyond 800 nm, and some cases were even tested far into the infrared. While some runs may statistically

prove be at the 700-800 nm wavelengths, having only slightly better correlation than at 550 nm, the runs having initial correlation under the total average and showing improvement beyond 800 nm, gives rise to the fact that there are other factors besides wavelength that are causing the error.

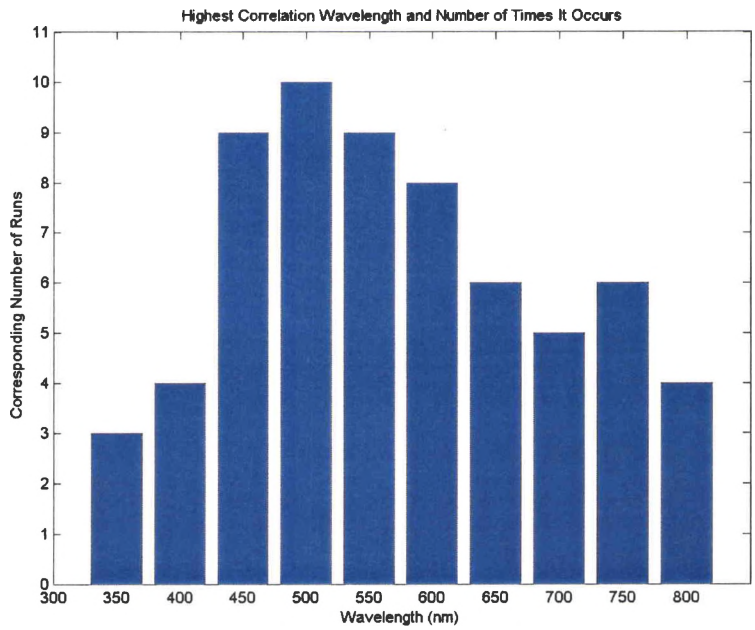

Figure 6.22 Correlation Histogram

Of the 65 runs having initial correlations above the average of 90.3%, the most frequently occurring wavelengths ranged from 450 to 550 nm. Figure 6.22 shows that there is good statistical representation over the group and that the value being used of 532.8 nm as the simulated laser beam wavelength is acceptable for the purpose of the simulation of the recording experiment.

It would be of relevance here to make an observation about different wavelengths traveling through the same medium. Recall from chapter 4 that the phase of a wave

passing through a medium depends on three variables, the refractive index of the medium, the length traveled in the medium, and the wavelength of the wave passing through the medium. If a single run is looked at with varying wavelengths, the corresponding output phases would differ. A sample of this can be seen in figure 6.23 where set 4, image 10 was run varying the wavelengths through the typical visible wavelength band.

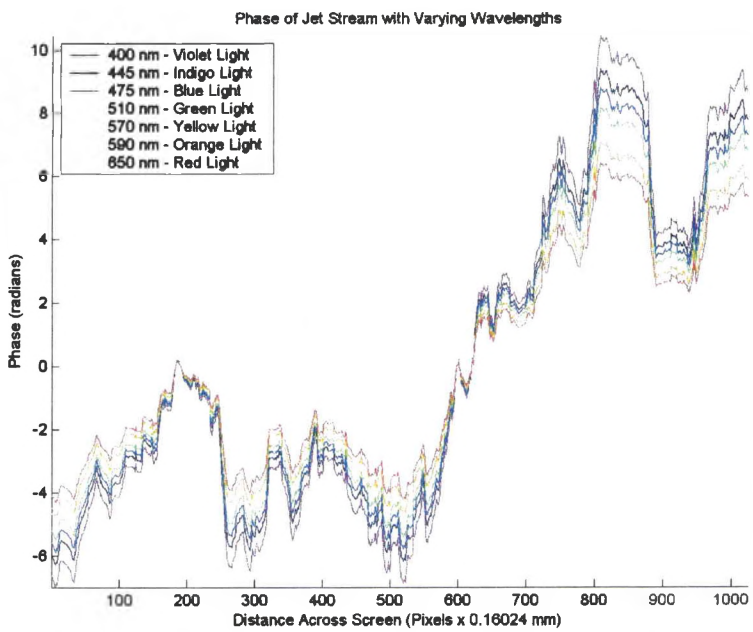

Figure 6.23 Phase of Varying Wavelengths in Same Medium

The reason that this occurs is as follows. Recall from chapter 3, the change of phase experienced by a wave of the form,

$$\Delta\phi = k(z - vt) \tag{6.5}$$

Where the wavelength is represented in the propagation constant, k, from equation 3.42, of $k = 2\pi / \lambda$, the distance that the wave travels is the medium is of the length, z, and the parameters of v and t are the velocity and the amount of time of which the wave passes through and spends time in the medium. It is here that we find that wavelength itself is not an isolated parameter, but that the velocity and time spent in the medium are related to index of refraction, which is a function of wavelength. The reason this occurs is because as a wave passes through the medium, some of the photons will collide with atoms and excite electrons to a higher energy state. Effectively, electrons absorb the photons, raising their energy levels, and then release the energy in the form of other photons. Depending on the resonant or natural frequency of the atom and the frequency of the incoming wave, the emitted photon will have changed phase when compared to the unaffected photons.

This is not only explains why the change of overall phase occurs in a single wavelength, but also explains why the different wavelengths have different phases as seen in figure 6.23. Each wavelength has a different frequency and is absorbed and released at different rates. These different rates cause a separation of phase between wavelengths, known as dispersion. Similar to the propagation constant, k, dispersion has a 1 over λ relationship. Hence, as the wavelength increases, the propagation constant and the dispersion factor both decrease, and as of such, have an overall effect of a decreasing the change of the phase factor.

6.4.4 Output Correlation with Rotation

The final validation of the simulation is testing for results of rotation. Similarly to the evaluation of vary the wavelengths, the runs were put through a loop of varying the rotation of the image from -2° to 2° in the x-z plane in increments of 0.25°. This rotation is demonstrated visually in figure 6.24, and represents the simulation of any possibility of the beam coming into the flow at an angle up to two degrees.

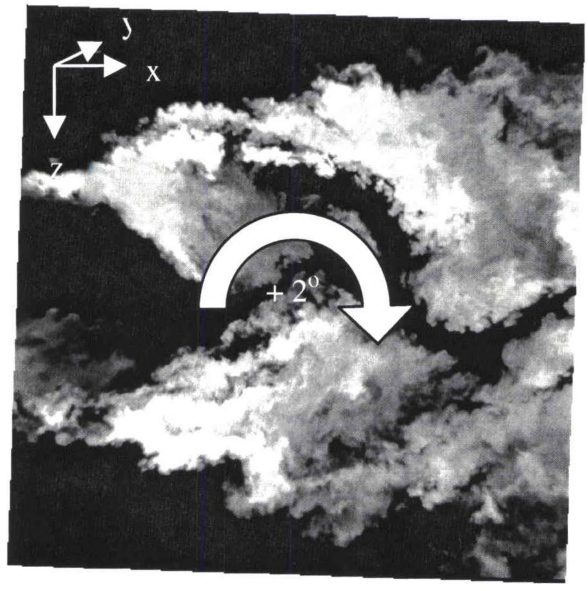

Figure 6.24 +2° Image Rotation Visualization

Each run image was rotated from -2° to 2° in the x-z plane in increments of 0.25° and then a correlation was performed to seek out the effect that rotation had on correlation performance. Figure 6.25 gives and example of one of the trials that this was performed upon.

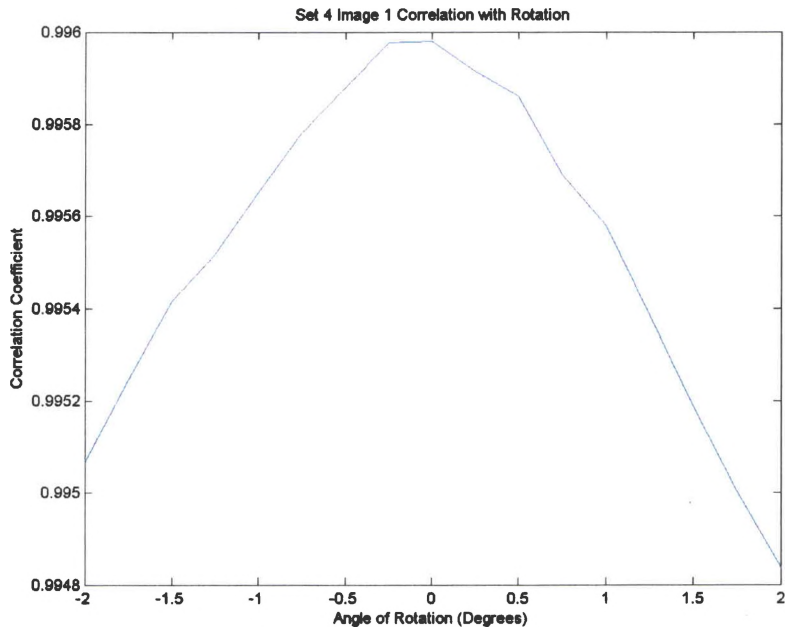

Figure 6.25 Sample Correlation with Varying Rotation

While the correlation of the wavelengths of section 6.4.3 showed a distribution of the best correlation at different wavelengths, the correlation varying the rotation showed one of three things. Either the correlation had a best fit at 0° rotation, or there was better correlation at the maximum rotations of + 2° or - 2°. However, it is of significance to note to the scale of improvement of the correlation coefficient for rotation. Note the scale of improvement in figures 6.21 and 6.25. For the wavelength improvement, the correlation coefficient changed approximately 5% from maximum to minimum, whereas the correlation coefficient for the rotation from maximum to minimum for the trial changed only by a factor of 0.1%. While significantly small for this case, no trials where rotation occurred varied by more than a change of improvement of 0.01 in the correlation coefficient, or improvement of better than 1%. As a result, it is deemed that rotation in small degrees does not play a significant role in varying the phase factors.

6.5 Beam Characteristics Beyond the Sensor Position

Now that the code has been verified in performance, it can be used to make assumptions about the laser beam down stream. While there were no measurements made to validate this performance at the distances that will be discussed, because the correlations with phase are high, these predictions are assumed to be within a reasonable range actual beam performance. Chapter 7 will discuss possibilities of further research which could validate down stream performance.

Of the many possible parameters that could be explored, it is the focus of this section to analyze the beam radius and beam center location with respect to an undisturbed beam propagated to the distance of two meters, fifty meters, and one hundred meters. The reasoning behind choosing these three distances of propagation are as follows. Two meter propagation was chosen to inspect the beam quality at a distance close to the original position of the sensor. The distance of one hundred meters was chosen to represent a distance far down stream. This distance needed to be far enough to allow for distinguishable character tics but couldn't be too far as to run into issues of the phase and intensity of the beam running back into itself as a byproduct of the reflective boundaries due to the method of correlation. A distance in between the extremes, the distance of fifty meters, was chosen to represent a distance half way between to inspect any phenomena which may had taken place and not realized in the one hundred meter propagation.

Basically, as the beam propagated, the phase constructively and destructively altered the beam. Figure 6.26 gives an idea of the types of change that can occur. This

figure represents two cross sections of intensities of a laser beam. The first is the undisturbed beam; the second is the beam at some distance down steam after passing through the high speed flow.

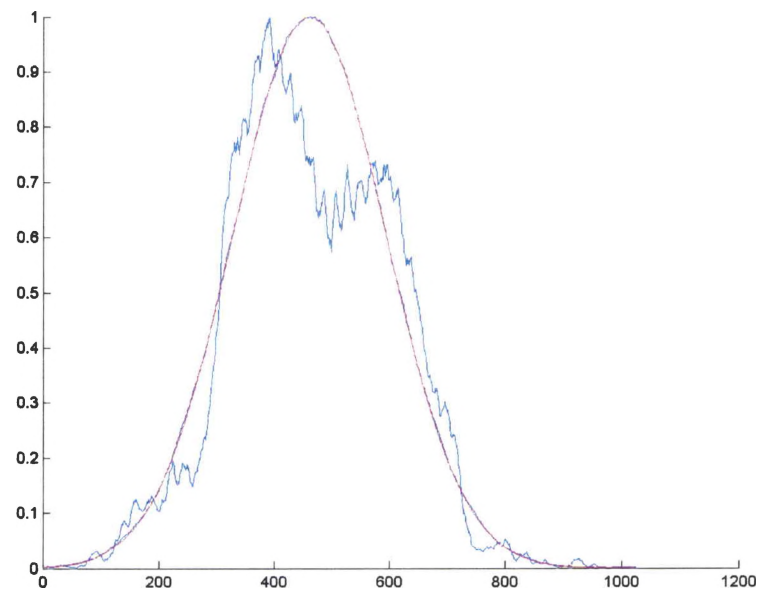

Figure 6.26 Sample Cross Section of Intensities Down Stream

Note in figure 6.26 that the beam is beginning to show a prominent peak to the left of center of the original beam. As a result, if a Gaussian beam profile is fit to the disturbed beam, it would show that the beam is beginning to travel to the left of the original beam and beginning to spread out to degree slightly greater than the undisturbed beam. It is this type of analysis that was accomplished for the simulated runs performed in this research.

6.5.1 Gaussian Beam Fitting

The first step on finding the beam radii and center locations was to perform a Gaussian beam fit to a cross section of the simulated beam. This was performed using a Matlab routine developed by Dr. Scott Harris. This simple routine uses the basic Gaussian radial distribution curve, given in equation 6.6.

$$I(r) = I_o e^{\left(\frac{-2r^2}{w^2}\right)} \tag{6.6}$$

Where I(r) represents the radial intensity of the beam

I_o is the beam amplitude

r is the distance from beam center

w is the $1/e^2$ beam waist

Each given image from the recording experiment was propagated out to a distance of two, fifty, and one hundred meters and the Gaussian curve fit routine was performed. Also included in the fitting process was a determination of the chi squared value of the fit. The basis for performing the chi squared value was to determine the variance in the curve fitting of the Gaussian shape. An example of the curve fitting accomplished can be seen in figure 6.27.

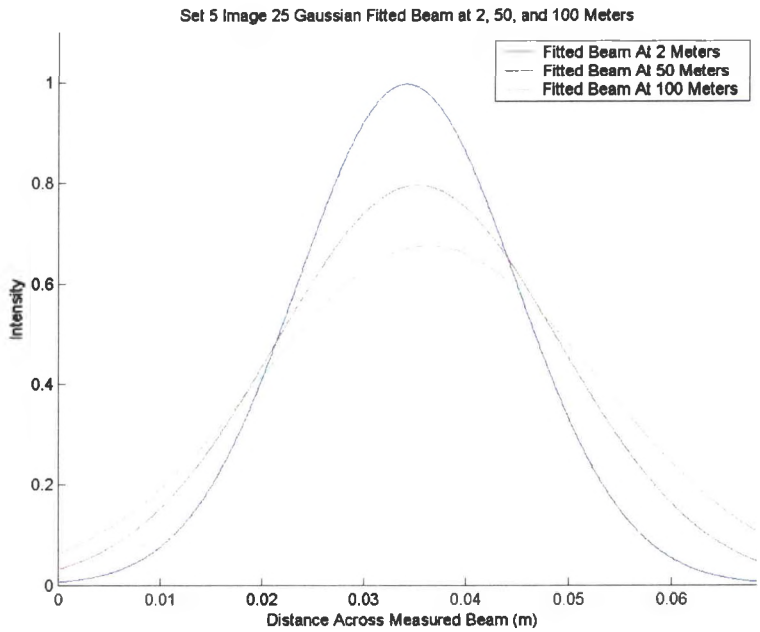

Figure 6.27 Gaussian Curve Fitting at 2, 50, and 100 meters

Figure 6.27 gives the curves found using the set 5, image 25 jet stream image and performing the conversion to phase screen discussed in chapter 4, then propagated out to distances of two, fifty, and one hundred meters and then performing the curved fitting routine on. It is noticeable form the figure, that for this case, the beam is indeed spreading out as it propagates as well as shifting to the right. The analysis of all 5 sets were performed and stored for use for the determination of the beam center and beam radii and comparison to undisturbed beam.

6.5.2 Beam Centers Down Stream

The analysis of the beam centers showed a variety of shifts from an undisturbed beam. These shifts ranged to +/- 4 mm from an undisturbed beam. Figure 6.28 presents

the beam centers for all valid sets propagated to two, fifty, and one hundred meters. Note that the original and undisturbed beam centers had a location of 34.197 mm.

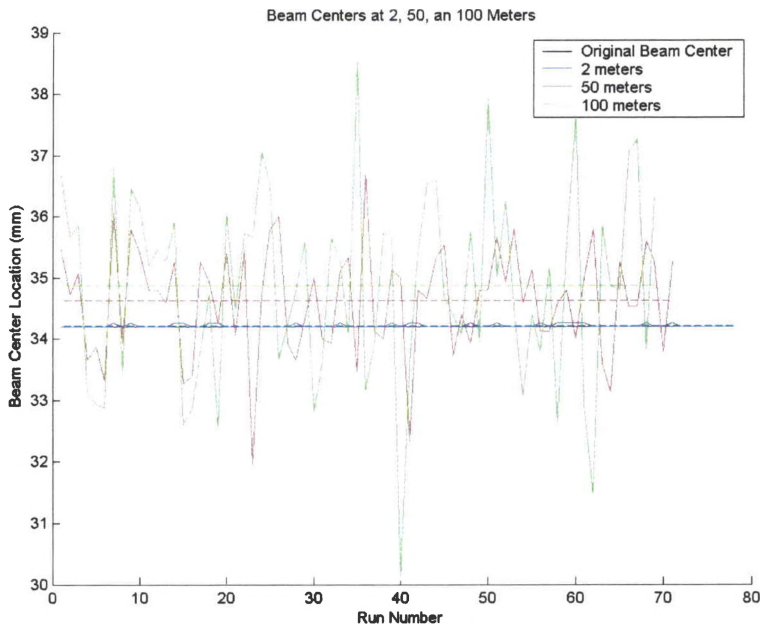

Figure 6.28 Beam Centers at 2, 50, and 100 Meters

Analysis of beam centers show that for majority of the runs, the beam is shifted to the right, when looking down on the beam as it travels in the down direction, as compared to the undisturbed beam. This should come as no surprise given the geometry of the high speed flow. Basically, as the beam comes in, it hits the angled mixing layer is directed to the right. Even if the two mixing layers were the same and the core layer was symmetrical, and cancel out any tilt effects, the beam will have shifted to the right as it propagated through the stream. What does come out as a small surprise, however, is that in some cases, the beam shifts back to the left. This comes as a result of the fact that the core and mixing layers are not symmetrical with respect to the axis and that an infinite

amount of boundary layer angles can occur as a result of the turbulent mixing in the high speed flow.

The average shift and standard deviation of each propagation distance can also be seen in figure 6.28. For the two meter propagations, which one would expect to be low, the amount of shift was 0.017 millimeters from an undisturbed beam with a standard deviation of 0.029 millimeters. Here, a positive amount denotes that the beam shift was to the right while looking down on the beam propagating down the direction. The fifty meter propagation showed an average beam shift of 0.435 millimeters with a standard deviation of 0.884 millimeters. The one hundred meter propagation showed an average beam shift from an undisturbed beam of 0.671 millimeters with a standard deviation of 1.606 millimeters.

While these numbers for shift through a flow may seem low, one must keep in mind that the total propagation distance in the flow is only approximately five centimeters. Coinciding with this is the fact that change of refractive index is only 0.0007 and one can imagine that the total phase change would be relatively small in comparison. Also, the flow is, on average, pretty much symmetrical, so any large angles produced as the beam initially enters the flow will be angled back upon exiting the flow. The flow boundaries, however, are random, and the data shows this. In fact, in some cases the beam shifted as much as four millimeters. However, it shifted back the other direction in another trial to a degree of four millimeters, thereby forcing the average of the two to be zero. So while the beam shift is not so much apparent for this application, where one would see the degradation of the beam would be in the difference of beam spread verses an undisturbed beam.

While one might assume that the amount of shift at one hundred meters would simply be twice the amount of the shift at fifty meters, this is an incorrect assumption. As the beam propagates, the phase will constructively, as well as destructively, shift the beam intensity around as the beam propagates. The shift will not likely be linear with distance; however the overall standard deviation, given enough trials, would entail the distribution that should show a linear transition. Here, the standard deviation at one hundred meters is just under twice the amount of the standard deviation at fifty meters.

Figures 6.29 and 6.30 are included here to allow for an improved view of the beam centers, averages, and standard deviations for both the fifty and one hundred meter propagations. Note the undisturbed beam center is also represented in the figures.

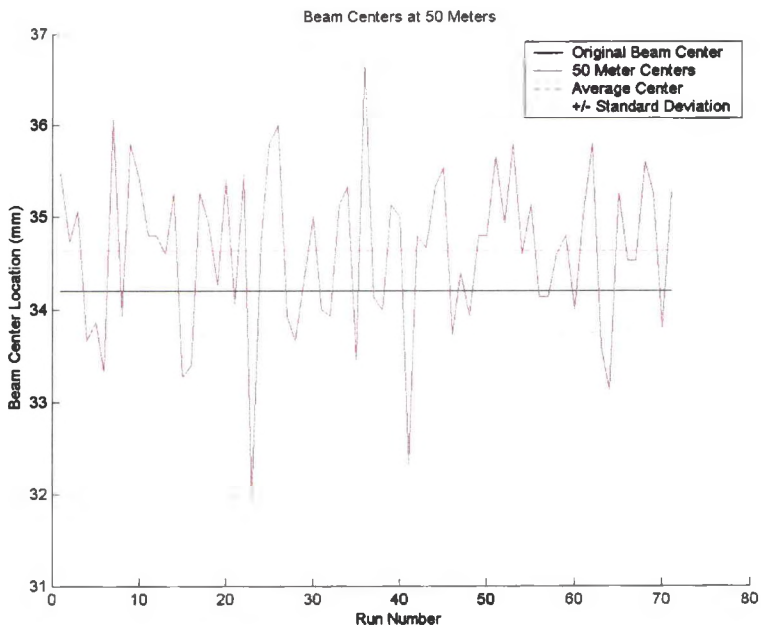

Figure 6.29 Beam Centers at 50 Meters

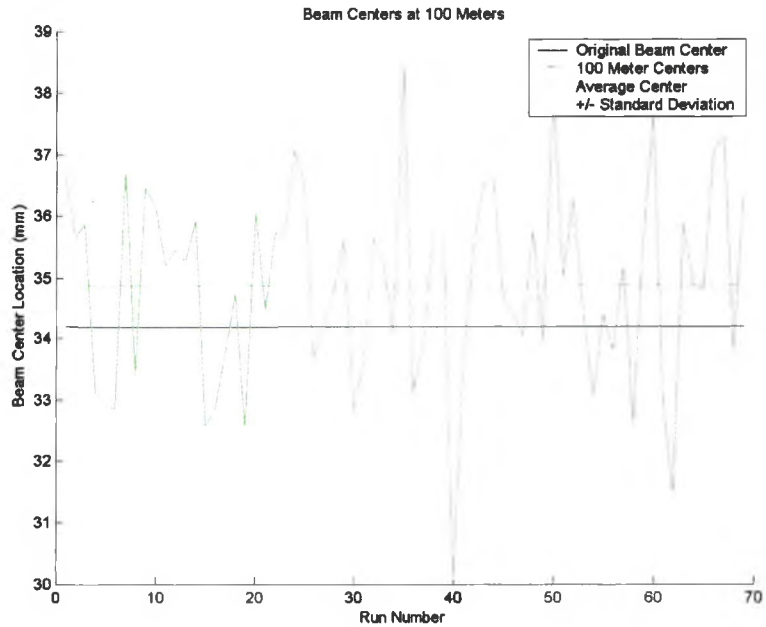

Figure 6.30 Beam Centers at 100 Meters

6.5.3 Beam Radii Down Stream

The final step in the analysis performed of the characteristics beyond the sensor in this research is the exploration of the beam radii. Recall from chapter 2, the laser used in the recording experiment was a collimated beam. As a result, an undisturbed beam would have beam spread only as a result of diffraction. The propagation tools used in the simulation take this diffraction into account; therefore, any effect introduced besides the diffraction can be directly measured by subtracting out the beam spread from diffraction.

To accomplish this, a series of simulations were performed similar those completed for section 6.5.2, the determination of beam centers. Here, however, the recorded values from the Gaussian curve fits were the $1/e^2$ beam radii. The same two, fifty, and one hundred meter propagations were performed and the radius of the beam for

each run was recorded. The results can be seen in figure 6.31 below along with average radii occurring at two, fifty, and one hundred meters.

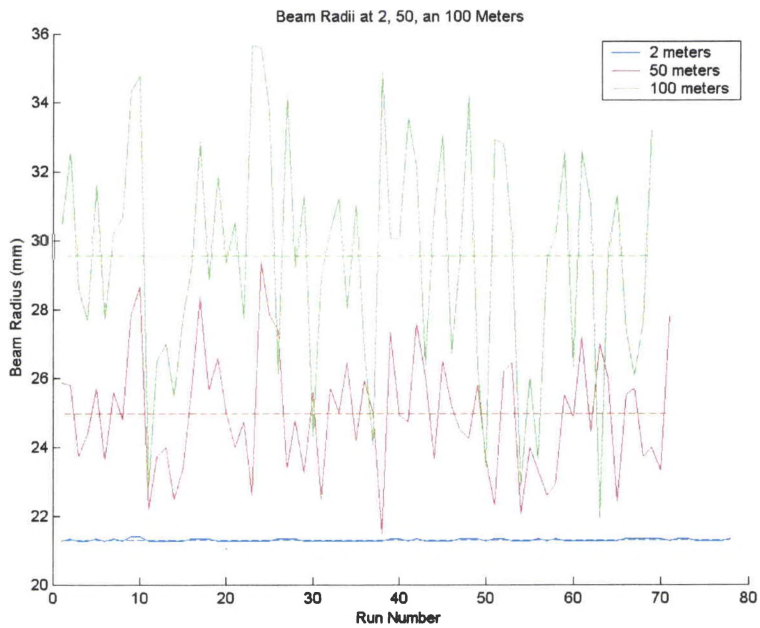

Figure 6.31 Beam Radii at 2, 50, and 100 Meters

While it is clear in figure 6.31 that there is a dramatic difference between the beam radii as a result of passing through the high speed flow, with beam radii ranging from approximately 22 to 36 mm at the one hundred meter propagation, it is not clear how much different the beam spread is from normal spread due to diffraction. To help quantify the amount of beam spread resulting from the high speed flow, the percent difference, equation 6.3, shown here again for reference as equation 6.7, is used.

$$\%Dev = \frac{(X_m - X_t)}{X_t} \times 100\% \qquad (6.3)$$

Here, the true value, X_t, is the beam radius of the undisturbed propagated beam and the measured value, X_m, is value of the beam radius propagated the same distance after passing through the high speed flow. The percent difference for all five sets with valid runs can be seen in figure 6.32, along the average percent difference.

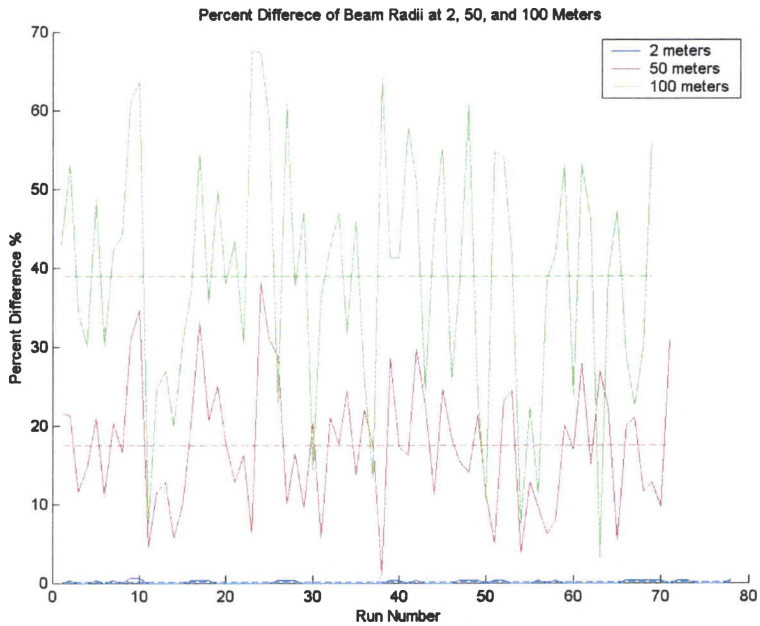

Figure 6.32 Percent Difference of Beam Radii at 2, 50, and 100 Meters

To aid in distinguishing the beam radii at fifty and one hundred meters, figures 6.33 and 6.34 are presented along with respective average values and standard deviations.

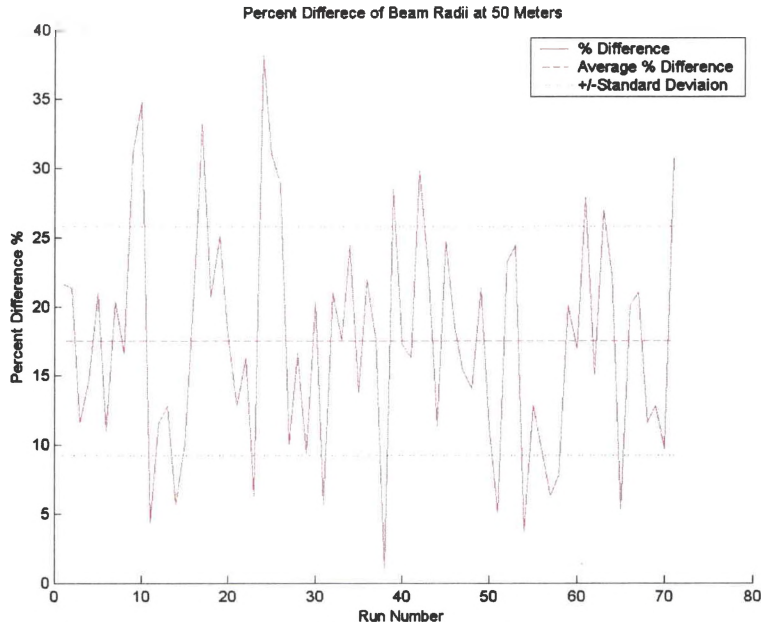

Figure 6.33 Percent Difference of Beam Radii at 50 Meters

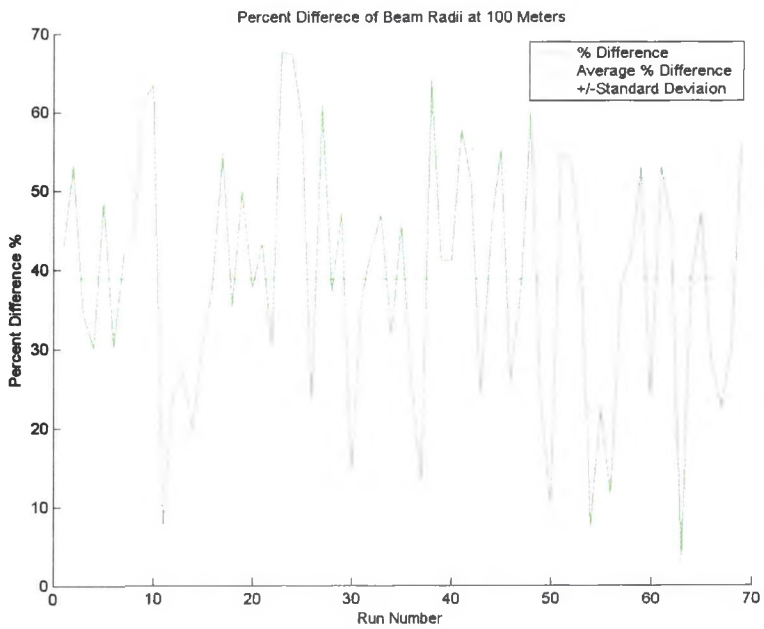

Figure 6.34 Percent Difference of Beam Radii at 100 Meters

The simulated output of the previous figures shows the following results. For a propagation range of two meters, the average difference in beam radius from the high speed flow as compared to an undisturbed beam is 0.026 millimeters with a standard deviation of 0.036 millimeters. This corresponds to an average percent difference of 0.12 % with a standard deviation of 0.17 %.

At a propagation distance of fifty meters, the average difference in beam radius from the high speed flow as compared to an undisturbed beam is 3.725 millimeters with a standard deviation of 1.757 millimeters. These numbers correspond to an average percent difference when propagating to fifty meters of 17.52 % with a standard deviation of 8.26 %.

At a propagation distance of one hundred meters, the average difference in beam radius from the high speed flow as compared to an undisturbed beam is 8.282 millimeters with a standard deviation of 3.371 millimeters. These numbers correspond to an average percent difference when propagating to one hundred meters of 38.95 % with a standard deviation of 15.85 %.

CHAPTER 7

CONCLUSIONS

The research accomplished in this thesis was performed to fulfill an objective to aid in the study of laser beam propagation and the aberrations caused in the beam by a high speed flow. This joint DAGSI study involved the University of Dayton[6], the Ohio State University[7], the Air Vehicles Directorate[8] of the Air Force Research Laboratories, and the Sensors Directorate[9] of the Air Force Research Laboratories.[68] While the overall joint research pertained to the prediction and control of aero-optic aberrations, the work done in this thesis focuses on modeling and simulation to predict the amount of aberration introduced into the laser beam as it passed through a turbulent flow.

First, a survey of existing beam propagation software capable of analyzing laser beam properties before, during, and after the propagation through a turbulent layer. Several methods of beam propagation were explored for use in the simulation. Upon development of the finalized code, a series of analytical tests were performed to verify the code's ability to accurately simulate similar conditions to those associated with the OSU/UD recording experiment.[17] The conditions of the experiment were documented along with the recorded images of the high speed flow and corresponding wavefront information for several trials. The images were then used as inputs to the computer code and a set of simulated output phases was obtained. These output phases were then

correlated against the measured phases in the validation of the simulation. Simulations were then performed to explore the laser beam characteristics downstream.

7.1 Simulation Results

The simulation designed for the propagation of a laser beam through a high speed flow was put through a series of tests to validate the output information. These tests, documented in detail in section 6.1, included verification by visual inspection, verification by trial, and verification by calculation. Known outputs could be determined by the use of a designed cube. This cube not only provided the means of having a known output, but also allowed for the comparison of the beam propagation methods as well as providing a method for validating the rotation.

Upon the successful completion of the validating tests, it was determined that the method most effective for the use of beam propagation was the Fast Fourier transform. This method was then used to correlate the results of the recording experiment. Of the 129 valid runs, the average correlation was determined to be 90.3%. Possible errors as a result of shift, rotation, and wavelength variance were also explored. The error from shifts was reduced by finding the best fit value of doing a row by row comparison. This shift, however, occurred no more than +/- 0.658 cm in any one direction for any one run. A histogram of the wavelength variation was accomplished in section 6.4.3 revealed that the method of determining the phase was correct, lending credit to the representation of the mixing layers discussed in section 4.2. Error from rotation proved to be negligible when correlation was computed as a function of rotation. The overall correlation between the simulated and the measured phases not only proved that the representation of

the mixing layer was appropriate, but also the suitability of using a single phase screen to represent the turbulent flow.

Finally, the beam characteristics down stream at two, fifty, and one hundred meters were explored. Here, each valid run was propagated to the stated distance, a Gaussian beam fit was accomplished to find the beam center and the beam radius, and a comparison to an undisturbed ideal Gaussian beam was performed. This comparison allowed for the determination of the degree to which the high speed flow caused the beam to spread and shift. The average shift and standard deviation of each propagation distance was 0.017 millimeters with a standard deviation of 0.029 millimeters for a two meter propagation, 0.435 millimeters with a standard deviation of 0.884 millimeters for a fifty meter propagation, and 0.671 millimeters with a standard deviation of 1.606 millimeters for a one hundred meter propagation. The amount of beam spread in the radius for a an undisturbed beam traveling the same distance was 0.026 millimeters with a standard deviation of 0.036 millimeters at a two meter propagation, 3.725 millimeters with a standard deviation of 1.757 millimeters at a fifty meter propagation, and 8.282 millimeters with a standard deviation of 3.371 millimeters at a one hundred meter propagation. These results are discussed in depth in sections 6.5.2 and 6.5.3 respectively.

7.2 Improving Simulation Performance

While the simulation had a performance in correlation above 90%, there is still room for improvement. These improvements could be considered items of future work as a continuation of this research.

First, the overall inputs into the model could use a better investigation. The inputs used in the simulation, discussed in section 4.1 were cross section images of the condensation as the cold air dispersed from the jet nozzle. These images could be improved by taking a corresponding cross sectional image across the beam. This type of imaging could allow for a representation of what happens to the flow as the beam propagates across it. One way to perform this would be the full three dimensional mapping of the flow. Ultimately, this could be accomplished by reducing the amount of seeding in the flow and perform an analysis using a Rayleigh scattering technique. This method was used in the determination of flow structure in the shear layers of high Reynolds number jets by Dimotikis in 2001.[67] This ultimately would provide a three dimensional map of the refractive indices of the flow that could be then be substituted into the simulation for representation of the flow.

If the recorded inputs could not be changed, the next step would be to improve the mixing layer representation. While the current methodology of averaging the core and ambient layers proved to be acceptable, being able to better evaluate the characteristics of the flow would eliminate the cases where the correlation was poor. This could be as simple as performing an investigation into varying the mixing layer into several smaller layers and varying the size, or as complicated as performing an in-depth analysis of turbulent flow and using mathematical models to determined the flow characteristics. Here, several series of runs could be accomplished and then compared statistically to the recorded out put beam phase and other characteristics.

The beam characteristic predictions could also be explored. Here, the grid size around the simulated beam could be expanded out to larger size, allowing for propagation

to further distances. A follow on recording experiment, similar to the one performed as a precursor to this research, could be accomplished entailing a sensor at a further distance to provide data for correlation with simulations.

Ultimately, an experiment from an aircraft where the beam is propagated many hundreds of meters could be done. While the results presented here could possibly aid in the simulation of such and experiment, another model investigated for this research, WaveTrain, discussed in section 5.2.4 may also prove to be beneficial, as it was designed for long range propagation. A study such as this would aid in the exploration of military or sensing applications of directed energy beams from aircraft.

7.3 Significant Contributions

Of the work accomplished in this research is it of note to mention the significant contributions, not only achieved by the direct research involved in this thesis, but also the work that contributed to it. First, the collaborative of effort of the University of Dayton, the Ohio State University, and both Air Vehicles and Sensors Directorates of the Air Force Research Laboratories needs to be acknowledged. Without the effort put forth by all parties, the project in its entirety would not have been able to be accomplished.

Jeff Widiker of the University of Dayton and Brian Thurow of the Ohio State University, backed by their advisors, Dr. Bradley Duncan and Dr. Mo Samimy, respectively, whose work was vital in the recording experiment to set up the high speed flow as well as record the wavefronts using the Shack-Hartmann wavefront sensor. These efforts provided the inputs and measured outputs vital for the work accomplished entailed in this thesis.

My work, under both the University of Dayton, backed by Dr. Andrew Sarangan, as well under the Air Force Research Laboratory's Sensor Directorate, produced a valid simulation vital not only to the correlation of the measured data, but also in providing verification for the simulation, as well as providing a complete representation of the experiment in determining the setup and prediction of laser beam propagation through a high speed flow.

APPENDIX A

BEAM PROPAGATION CODE

```
% Beam Propagation and Comparison Code For Thesis

%  Luke Borntrager
%  University of Dayton

% 2003-2004

% This MATLAB code uses a jet stream image from the March Recording experiment,
% creates a laser beam, alters the image into a phase screen, propagates the beam through
% the beam, then propagates beam to any set distance.  Beam characteristics are
% investigated throughout the code and allows for several code verifications.

% clearing all variables and figures

    clear
    close all

% changing format of numbers to long to show account of very small
% changes in index of refraction

    format long

% Inputting a set to work with

  Set=input('Pick a set to work with (1 - 5):');

% performing a series of checks to chose the image numbers

  if Set == 1
    Image=input('Pick an image to work with (1-25):');

    while Image < 1 || Image > 25
      disp('Please pick a number between 1 and 25')
      Image=input('Pick an image to work with :');
    end

  elseif Set == 2
    Image=input('Pick an image to work with (1-49):');

    while Image < 1 || Image > 25
      disp('Please pick a number between 1 and 49')
      Image=input('Pick an image to work with :');
    end
```

```
elseif Set == 3
   Image=input('Pick an image to work with (1-24):');

   while Image < 1 || Image > 24
      disp('Please pick a number between 1 and 24')
      Image=input('Pick an image to work with :');
   end

elseif Set == 4
   Image=input('Pick an image to work with (1-25):');

   while Image < 1 || Image > 25
      disp('Please pick a number between 1 and 25')
      Image=input('Pick an image to work with :');
   end

elseif Set == 5
   Image=input('Pick an image to work with (1-25):');

   while Image < 1 || Image > 25
      disp('Please pick a number between 1 and 25')
      Image=input('Pick an image to work with :');
   end

else
   while Set < 1 || Set > 5
      disp('Please pick a Set numbered 1 - 5:')
      Set=input('Pick a set to work with :');
   end
end

% Saving the file for comparison

   savefile = ['C:\Documents and Settings\borntrla\Desktop\Thesis\Luke\Final Codes\Results\Compare
     Phase\Set ',num2str(Set),'\Set',num2str(Set),'_Information.mat'];
   save(savefile)

% Calling in the proper image and measured phase

   I=imread(['C:\Documents and Settings\borntrla\Desktop\Thesis\Luke\OSU Modified
     Images\Set',num2str(Set),'_Image',num2str(Image),'_Image.jpg'],'jpg');

% For the verification process, instead of the turbulent image being called in
% the two-tone cube was called in.  This was done under separate coding

% Convert image to a double array with maximum intensity of 1.

   I=rgb2ind(I,gray);
   close
   I=double(I);
   I=I-min(min(I));
   I=I/max(max(I))*1;


% Viewing the Image
```

```
    figure
    imshow(I)
    title(['Set ',num2str(Set),' Image ',num2str(Image)])

% Beginning Calculations of the phase

% Using imfill to fill in "holes" of the ambient and core layers

  I=imfill(I);
  I=abs(I-1);
  I=imfill(I);
  I=abs(I-1);

% Finding size parameters for calculations

  [Height Width]=size(I);

% Currently have values from 0 (black) to 1 (white)
% setting all non zero values (mixing layers) equal to .5.

  for i=1:Height;
    for j=1:Width;
      if I(i,j)>0
        I(i,j)=.5;
      else
        I(i,j)=0;
      end
    end
  end

%Examining image for verification

  figure
  imshow(I)
  title(['Set ',num2str(Set),' Image ',num2str(Image),' values of 0 to .5])
  vals=impixel

% Subtracting .25 from image makes the mixing layers positive and the
% the ambient and core layers negative.  Then loop to make all negative
% values equal 0 and all positive values equal to 1.

  I2=(I-.25);

  for i=1:Height;
    for j=1:Width;
      if I2(i,j)>0
        I2(i,j)=0;
      else
        I2(i,j)=1;
      end
    end
  end

% Verification check of image
```

```
figure
imagesc(I2)

% BWSelect is then used starting at the "top" and "bottom" of the turbulence
% image to choose the ambient areas of 1 and the rest equal to 0.

AmbientBot=BWselect(I2,1,1);

AmbientTop=BWselect(I2,1,Height);

% Finally the ambient (equal to one), core (equal to zero), and mixing
% layers (equal to .5) are all combined back together into one image.
% *Note that the image is then turned into a negative(ambient=0,mix=.5,
% and core=1) to allow for changes to layers below.

I=AmbientTop+AmbientBot+I;
I=abs(I-1);

% Visual verification

figure
imshow(I)
vals=impixel


% This section here allows for any rotation to be accomplished
% During the verification or rotation, this code was implemented
% in a loop in separate code.  It is added here in the final code
% to show the process of rotation.

% Asking the user if they want to do any rotation to the image
% image is rotated using imrotate and then must be cropped back to
% the original size.
% Also allows for visual check of rotation

RotateImage=input('Do you want to rotate the image? (Pick 1 for yes)');

if RotateImage==1
   Angle=input('How many degrees do you want to rotate the image?');
   J=imrotate(I,5,'bilinear');

   [Heightr Widthr]=size(J);
   I=imcrop(J,[(Heightr-Height)/2 (Widthr-Width)/2 (Height-1) (Width-1)]);

   SeeRotate=input('Do you want to see the rotated image? (Pick 1 for yes)');
   if SeeRotate==1
      figure
      imagesc(I)
      vals=impixel
   end
end


% Note because of rotation, not all pixels are represented equally an
% sections of zero are averaged into the mixing and core layers.  To fix
```

```
% this the imfill function is used again.

   I=imfill(I);

% Now plugging in the indices of refraction in each of the layers where
% ambient air = n1, core = n2 and mixing layer = nMix = (n2+n1)/2

   Tamb=286; % Measured temperature at recording experiment
   Pamb=29.66*3386.4; % Measured pressured converted from mm of Hg to N/m2
   Pamb=Pamb-Pamb/(287*T)*9.81*276; % Change of pressure to account for elevation
                                    % difference of Scott Field vs. where pressure was recorded

% Solving for temperature of core based on To/T=1 + [(k-1) / 2 ] * M^2

   k=1.4; % specific heat of ideal gas
   M=1.28;% speed of jet stream in Mach Number

   Tcore=Tamb/(1+((k-1)/2)*M^2);

   Pamb=p;

% Need to solve for Pressure of core based on Po/P=(1 + [(k-1) / 2 ] * M^2)^(1/(k-1))

   Pcore=Pamb/((1+((k-1)/2)*M^2)^(1/(k-1)));

% a=eo*(nlight^2-1)/ngas
% ngas=p/(kB*T)

   eo=8.854e-12;
   kB=1.381e-23;
   nlight=1.000293;
   ngasambient=Pamb/(kB*Tamb);
   ngascore=Pamb/(kB*Tcore);

   aabient=eo*(nlight^2-1)/ngasambient;
   acore=eo*(nlight^2-1)/ngascore;

   aair=1.70625e-40;
   n1=(1+(ngasambient*aair/eo))^(1/2);
   n2=(1+(ngascore*aair/eo))^(1/2);

   nMix=(n1+n2)/2;

% Plugging in refractive indices n1, n2, and nMix into the image

   for i=1:Height
      for j=1:Width

         if I(i,j)==1
            I(i,j)=n2;
         elseif I(i,j)==.5
            I(i,j)=nMix;
         else
            I(i,j)=n1;
         end
      end
```

```
end

% Checking Image to see if proper values are substituted in

figure
imagesc(I)
axis square
axis tight
colormap gray
vals=impixel

% Starting the calculations for the laser.

% Making sure all units are in meters

m=1;
nm=1e-9*m;
mm=1e-3*m;
cm=1e-2*m;

% Actual height of screen is 2.6875 inches.
% total pixel size = Height = 1024

PixelHeight=(2.6875/1024)*2.54*cm;

% User designating the wavelength

lambda=input('Please designate a wavlength in nms (Measured data used 532.8 nm):');

% Note it was here that the wavelength was changed in increments of 50 nm
% from 300 to 800 nm to allow for the wavelength

% Calculating the optical path length

OPL=PixelHeight*sum(I);

% Calculating the phase from the opitcal path length

OPD=OPL-mean(OPL);

Phase=(2*pi/lambda)*OPD;

% Showing the phase of the turbulence screen

figure
plot(Phase,'r')
title('Phase for Turbulence Screen')
ylabel('Phase (radians)')
xlabel('Distance in pixels (*.16024 mm)')

% Starting the Laserbeam and incorporating the phase screen



% Will start out with plane wave, then plug in a Gaussian beam profile,
% then propagate it up the turbulence, plug the phase screen, propagate
```

```
% it to the length of the recorded wave, then determine the result.

% First need to convert the phase screen into a three dimensional cube
% then compress into a screen of no thickness

% Take the phase from above and make it into a screen

   PhaseSize=size(Phase);

   PhaseSheet=ones(PhaseSize(2),PhaseSize(2));

   for i=1:PhaseSize(2);
      PhaseSheet(i,:)=Phase;
   end

% Visual Check

   figure
   imagesc(PhaseSheet)
   colormap gray

% system specifications - beginning the laser

% Note here teh use of the LightPipes functions.
% They are designated by the "LP" in the name of the function.

   lambda=lambda;% defined above

   FieldSize=PhaseSize(2)*PixelHeight; % total screen size
   Field=LPBegin(FieldSize,lambda,PhaseSize(2));
   BeamIntensity=LPIntensity(1,Field);

% NOTE: The beam is a plane wave with zero phase.
% Now we form the Gaussian beam by letting the user define the beam
% parameters.  The laser beam width used in the experiment was 30 mm

   w=input('Please designate the beam radius (1/e)  in mm :');

% Note the radius of the beam in the recording experiment was 15 mm

   R=input('Please designate the beam intensity (use 1 as max):');

   LaserBeam=LPGaussAperture(w,0,0,R,Field);
   BeamIntensity=LPIntensity(0,LaserBeam);

% Showing the Gaussian beam intensity as a visual check, both 2D and 3D

   figure
   plot(BeamIntensity(PhaseSize(2)/2,:),'r')
   axis([0 1024 0 1])

   figure
   imshow(BeamIntensity)

%Input length of beam travel before propagation and then propagate
```

```
ft=1/3.28*m;

z1=input('Input beam propagation before beam in feet');

% Note the beam propagation in feet was done because the distance
% given in the recording experiment was in feet.  Could change to meters
% for continuity if necessary.  In the experiment distance was z1=3*ft;

LaserBefore = LPForvard(z1,LaserBeam);

BeamIntensityBefore=LPIntensity(0,LaserBefore);

% Checking intensity of beam before jet stream

figure
hold
plot(BeamIntensityBefore(PhaseSize(2)/2,:),'r')
axis([0 1024 0 1])

%Mixing the turbulent sheet with the beam

LaserAfterT=LPMultPhase(PhaseSheet,LaserBefore);

%Input length of beam propagation after turbulence and then propagate

z2=input('Input beam propagation before beam in feet');

% Note the recording experiment had z2=2.75*ft;

% Now propagating the laser after the jet stream

LaserAfter = LPForvard(z2,LaserAfterT);
BeamIntensityAfter=LPIntensity(0,LaserAfter);

% Showing the intensity before and after propagation

figure
hold
plot(BeamIntensityBefore(PhaseSize(2)/2,:),'r')
plot(BeamIntensityAfter(PhaseSize(2)/2,:),'k')
legend('Before','After')
pause(2)

% Performing phase calculation

CalculatedPhase=LPPhase(LaserAfter);
CalculatedPhase=LPPhaseUnwrap(1/(2*pi),CalculatedPhase);

% Showing phase

figure
plot(CalculatedPhase(PhaseSize(2)/2,:),'r')
axis([0 1024 -pi pi])

%Comparing the simulated phase with the calculated phase
```

```
load(['C:\Documents and Settings\borntrla\Desktop\Thesis\Luke\OSU Modified
  Images\Set',num2str(Set),'_wavefront',num2str(Image),'']);
assignin('caller',['Set',num2str(Set),'_Image',num2str(Image),'_Phase'],phase);

% Changing invalid phase to low number as to not crash the code.
% Trials without recorded phase were discarded before analysis.

HasZEROPhase=sum(sum(phase));

if HasZEROPhase == 0
  phase=phase+.001
end

% Setting the correlation to 0 along with the
% starting row and columns.

BestCorrelation=0;

Line=0;
StartedAt=0;
MeasuredPhaseSize=size(phase);

% Starting a loop to do a row by row comparison of the
% phase to find the best fit between the phases

for i = 7:21
  for j=1:14

    MeasuredPhase=phase(i,:);

    StartAt=round(135+16.385*j);
    CalculatedPhase=LPPhase(LaserAfter);

    CalcPhase=imcrop(CalculatedPhase2,[StartAt 1024/2 425 0]);

% Averaging the simulated phase to match the measured phase sizes

    numberofpixs=size(CalcPhase);
    nblocks=numberofpixs(2);
    nblocks=26;
    block_size=round(length(CalcPhase)/nblocks);

    for block=1:nblocks
      CalcuPhase(block)=mean(CalcPhase((block-1)*block_size+1:block*block_size));
    end

    MeasuredPhase=imfilter(MeasuredPhase,fspecial('average',[3 3]),'replicate','same');
    CalculatedPhase=imfilter(CalcPhase,fspecial('average',[3 3]),'replicate','same');

    Shift=mean(MeasuredPhase)-mean(CalcuPhase);
    CalcuPhase=CalcuPhase+Shift;

    Correlation = xcorr(CalcuPhase,MeasuredPhase,0,'coeff');

% Performing a visual inspection row by row of the correlation.
% Note this should be commented out if doing a loop as
```

```
% it creates a large number of figures

        figure
        hold
        plot(CalcuPhase,'r')
        plot(MeasuredPhase,'k')
        legend('Predicted Phase','Measured Phase')
        axis([0 26  -pi  pi])
        title(['Set ',num2str(Set),' Image ',num2str(Image),' Phase Comparison for Row ',num2str(i),'.'])
        text(17,2,['Correlation = ',num2str(Correlation)])
        hold off
        pause(.1)
        close

% Setting the best correlation value
% along with row and column number

        if Correlation > BestCorrelation
           BestCorrelation = Correlation;
           Line=i;
           StartedAt=StartAt;
        End

     end
  end

% Outputting the best correlation information

  display(['Best Correlation for Set ',num2str(Set),' Image ',num2str(Image), ' is
    ',num2str(BestCorrelation),' at line ',num2str(Line),' starting at pixel ',num2str(StartedAt),'.'])

% Recording the phase correlation information into a .mat file

  DefinePredicted=['Predicted_Phase_Set',int2str(Set),'_Image',int2str(Image),'=CalcuPhase'];
  eval(DefinePredicted);

  DefineMeasured=['Measured_Phase_Set',int2str(Set),'_Image',int2str(Image),'=MeasuredPhase'];
  eval(DefineMeasured);

  TotalCorrelation(Image)=Correlation;


    save(savefile,['Predicted_Phase_Set',int2str(Set),'_Image',int2str(Image),],['Measured_Phase_Set',int2s
    tr(Set),'_Image',int2str(Image),],'-append')

% plotting and recording the figures of the best correlations
% as well as more parameters of the correlation

  close all
  Average=mean(TotalCorrelation(1:18));
  AverageLine=ones(length(TotalCorrelation))*Average;
  Ninety=ones(length(TotalCorrelation))*.9;
```

```
figure(2)
hold
plot(TotalCorrelation,'b-d')
plot(AverageLine,'r')
%plot(Ninety,'g')
axis ([0 NumberOfImages 0 1])
legend('Correlation of Predicted and Measured','Average Correlation for Entire Set',4)
title(['Correlation for Set ',num2str(Set),])


h=figure(2);

saveas(h,['C:\Documents and Settings\borntrla\Desktop\Thesis\Luke\Final Codes\Results\Compare
  Phase\Set ',num2str(Set),' Total Correlation.jpg'])
saveas(h,['C:\Documents and Settings\borntrla\Desktop\Thesis\Luke\Final Codes\Results\Compare
  Phase\Set ',num2str(Set),'\Set ',num2str(Set),' Total Correlation.jpg'])
TotalCorrelationofSet=['Total_Correlation_Set',int2str(Set),'=TotalCorrelation'];
eval(TotalCorrelationofSet);
TotalCorrelationAverageLine=['Total_Correlation_Average_Line_Set',int2str(Set),'=AverageLine'];
eval(TotalCorrelationAverageLine);
TotalCorrelationAverage=['Total_Correlation_Average_Set',int2str(Set),'=Average'];
eval(TotalCorrelationAverage);


  save(savefile,['Total_Correlation_Set',int2str(Set),],['Total_Correlation_Average_Line_Set',int2str(Set)
  ,],['Total_Correlation_Average_Set',int2str(Set),],'-append')

% Note for the beam comparisons a distance the code above was used
% however, the propagation was changed to 2, 50, and 100 meters.

% The following functions were then used to find the Gaussian fit.
% The code outlined by the ******** is contributed to the help
% of Jeff Widiker and Scott Harris.

% Gaussian curve fitting code

%*****************************************

% This is Gaussian curve fitting code

% function [A] = gaussfind(yvalues)

[junk, length]=size(yvalues);
parameters(1) = max(yvalues)*sqrt(2*pi); % Gaussian Amplitude
parameters(2) = round(length/2);         % Beam Center
parameters(3) = round(length/20);        % FWHM
parameters(4) = 0;                       % DC offset

options= optimset('TolX',1e-1,'MaxIter',2e3,'MaxFunEvals',2e3,'TolFun',0.001,'Display','off');

A=fminsearch('gaussfit1d',parameters,options,yvalues) % Calls Guass fit function defined below


% function chi = gaussfit1d(param,yvalues)
```

```
% This function will take a image matrix (fold) and find the chi squared value of it fit to a
% Gaussian with parameters (param. param needs to be of the following orientation
% param = (gauss amp modulation, colum center, vert fwhm , row center, hor fwhm, DC offset)

   chi=0;
   [junk, length]=size(yvalues);% determine array size;

   for i=1:length     % number of rows
      chi= chi + ( (param(1)/(2*pi)^0.5)*exp( -(i-param(2))^2/(2*param(3)^2))+param(4) - yvalues(i) )^2;
   end

   chi = chi/(length)

%*****************************************

% The Gaussian fits were then recorded into .mat files similar to the above
% code but is not listed here.  These parameters where then called back in
% in separate coding and used in plots and calculations to show the different
%amounts of beam spread and beam shift.
```

REFERENCES

1. Hecht, Jeff, *Understanding Lasers: An Entry Level Guide*, IEEE Press, The Institute of Electrical and Electronic Engineers, Inc., New York, p15, (1992).

2. "Prediction and Control of Aero-Optic Aberrations", *Joint AFRL/DAGSI Research Proposal, Sensors Directorate, SN-12*, AFRL/DAGSI/SN-12/OSU-UD-AFRL (SN & VA), (19 April 2001).

3. Roggeman, Michael C, and David J. Lee, "Two-Deformable Mirror Concept for Correcting Scintillation Effects in Laser Beam Projection Through the Turbulent Atmosphere", *Applied Optics*, Vol. 37, No. 21, (20 July 1988).

4. Dayton Area Graduate Studies Institute, 3155 Research Boulevard, Suite 205, Kettering, Ohio, 45420.

5. "Measurement and Modeling of Aero-Optical Aberrations in Coherent Laser Radiation", The Dayton Area Graduate Studies Institute (DAGSI) Award Number SN-OSU-01-06.

6. University of Dayton, 300 College Park, Dayton, Ohio 45469-0245.

7. The Ohio State University, Columbus, Ohio 43210.

8. Air Force Research Laboratories, Air Vehicles Directorate, Building 45, 2130 8th Street, Wright Patterson AFB, Ohio, 45433

9. Air Force Research Laboratories, Sensors Directorate, Building 620, 2241 Avionics Circle, Wright Patterson AFB, Ohio, 45433

10. Widiker, Jeffry J., *Design & Application of a High-Speed Shack-Hartmann Wavefront Sensor*, Thesis, University of Datyon, (2003).

11. The Ohio State University Gas Dynamics and Turbulence Laboratory, Aeronautical and Astronautical Research Laboratory, 2300 West Case Road, Columbus, Ohio 43235

12. Kerechanin, C. W., M. Samimy, and J.H. Kim, "Effects of Nozzle Trailing Edge Modification on Noise Radiation in a Supersonic Rectangular Jet", AIAA Paper, 2002-0086, (2002)

13. 1.3 Mach Nozzle Picture, Courtesy of Luke Borntrager, taken March 19, 2004 at the Ohio State University Gas Dynamics and Turbulence Laboratory.

14. Time Averaged Jet Stream Images, Courtesy of Brian Thurow, the Ohio State University Gas Dynamics and Turbulence Laboratory.

15. The Ohio State University Gas Dynamics and Turbulence Laboratory website, http://rclsgi.eng.ohio-state.edu/~samimy/GDTL/GDTL.htm

16. Thurow B., M. Samimy, W. Lempert, S.R. Harris, J. Widiker and B. Duncan, "Simultaneous High-resolution Optical Wavefront and Flow Diagnostics for High-speed Flows", AIAA Paper 2003-3613, (2003)

17. Thurow, Brian, and Jeff Whitaker, March recording experiment data collection using Shack Hartmann sensor, the Ohio State University, Don Scott Field, March, 2003.

18. Elliott, G. S., Samimy, M., and Arnette, S. A., "A Study of Compressible Mixing Layers Using Filtered Rayleigh Scattering", AIAA Paper 1992-0175, (1992)

19. Flow Visualization Using High Speed CCD Camera, Image from data provided by Brian Thurow, the Ohio State University, from the March recording experiment.[17]

20. Widiker, Jeffry J., and Dr. Bradley D. Duncan, the University of Dayton, and Dr. Scott R. Harris, Air Force Research Laboratories.

21. Pollock, Clifford R., *Fundamentals of Optoelectronics*, Richard R. Irwin, Inc., Chicago, Illinois, p 13, (1995).

22. Silfvast, William T., *Laser Fundamentals*, Cambridge University Press, New York, New York, pp 9-11, (1996).

23. Schey, Harry M., *Div, Grad, Curl, and All That*, Third Edition, W. W. Norton & Company, New York, New York, Inside front cover, (1997).

24. Pollock, Clifford R., *Fundamentals of Optoelectronics*, Richard R. Irwin, Inc., Chicago, Illinois, p 18, (1995).

25. Griffiths, David J., *Introduction to Electrodynamics*, Second Edition, Prentice Hall, Upper Saddle River, New Jersey, p 26, (1989).

26. MathWorld.Wolfram.Com, Wolfram Research, "Partial Differential Equations, Wave Equation", http://mathworld.wolfram.com/WaveEquation.html, (2004).

27. Schey, Harry M., *Div, Grad, Curl, and All That*, Third Edition, W. W. Norton & Company, New York, New York, p122 (1997).

28. Weisstein, Eric, "Fourier Transform", http://Enciklopedija/math/math/f/f274.htm, (1999)

29. Bernard, Peter S., and James M. Wallace, *Turbulent Flow, Analysis, Measurement, and Prediction*, John Wiley & Sons, Inc, New York, New York, (2002).

30. Piquet, Jean, *Turbulent Flow, Models and Physics,* Springer-Verlag Berlin Heidelberg, Heidelberg, Germany, (1999)

31. Pollock, Clifford R., *Fundamentals of Optoelectronics*, Richard R. Irwin, Inc., Chicago, Illinois, p p23-24, (1995).

32. Andrews, Larry C., and Ronald L. Phillips, "Laser Beam Propagation through Random Media", SPIE Optical Engineering Press, Bellingham, Washington, pp 7-9, (1998).

33. Fox, Robert W., and Alan T. McDonald, *Introduction to Fluid Mechanics*, Fifth Edition, John Wiley & Sons, Inc., New York, New York, pp 600-605, (1998).

34. Thurow, B, M. Sammimy, W. Lampert, S.R. Harris, J. Widiker, and B. Duncan, "Simultaneous High Resolution Optical Wavefront and Flow Diagnostics for High-speed Flows," AIAA Paper 2003-3613 (2003).

35. Griffiths, David J., *Introduction to Electrodynamics*, Second Edition, Prentice Hall, Upper Saddle River, New Jersey, pp 340-345, (1989).

36. Podesta, Michael, *Understanding the Properties of Matter*, Second Edition, Taylor & Francis Inc., New York, New York, pp 17-18, (2002).

37. Podesta, Michael, *Understanding the Properties of Matter*, Second Edition, Taylor & Francis Inc., New York, New York, pp 313-321, (2002).

38. Chang, Raymond, *Physical Chemistry for the Chemical and Biological Sciences*, University Science Books, Sausalito, California, p 788, (2000).

39. Lide, David R., *CRC Handbook of Chemistry and Physics*, 84th Edition, CRC Press, Boca Raton, Florida, pp 10.163-10.177, (2003).

40. Hecht, Eugene, *Optics*, Second Edition, Addison-Wesley Publishing Company, Reading, Massachusetts, p 89, (1990).

41. ABCD Gaussian Beam Propagation Software, www3.sympatico.ca/dccote/abcd

42. KAGI, 1442-A Walnut Street #392-ZTPBerkeley, California 94709-1405

43. WinLase Laser Design and Beam Propagation Software, www.winlase.com

44. Laser & Computer Consulting, Im Moselsgrund 26, D-69118 Heidelberg, Germany

45. Future Laser Technologies, 5051 Alton Pkwy #76, Irvine, CA 92604

46. SCIOPT Enterprises , P.O. Box 20637, San Jose CA 95160 USA

47. OPTEC-IV, Optical system design software, www.sciopt.com/optec.htm

48. SIGRAPH-OPTIK, Integrated optics applications software, www.sciopt.com/caos.htm

49. PARAXIA-Plus, Laser system design software, www.sciopt.com/paraxpls.htm

50. WaveTrain, MZA Corporation Adaptive Optics Model, www.mza.com/wavetrain/index.html.

51. MZA Associates Corporation, 2021 Girard SE, Suite 150, Albuquerque, NM 87106-3140.

52. Airborne Laser Program, www.airbornelaser.com.

53. Air Force Research Laboratory Energy Directorate, Office of Public Affairs 3550 Aberdeen Avenue S.E., Kirtland AFB, NM 87117-5776, www.de.afrl.af.mil.

54. Simulink, Mathworks Block Diagram Modeling Architecture, www.mathworks.com/products/simulink.

55. Matlab, Mathworks Coding Environment, www.mathworks.com/products/matlab.

56. Tempus, MZA Corporation Block Diagram Modeling Architecture, www.mza.com/tempus/index.html

57. Atmospheric Compensation Simulation, ACS, SAIC Wave Optics Code.

58. Yet Another Propagation Simulation, YAPS, Phillips Laboratory Beam Propagation Software, linus.cast.uni-linz.ac.at/st/stims/yaps.html.

59. OKO Technologies, Reinier de Graafweg 300, 2625 DJ Delft, The Netherlands.

60. LightPipes, Beam Propagation Toolbox, www.okotech.com/software/lightpipes.

61. Goodman, Joseph W., *Introduction to Fourier Optics*, Second Edition, The McGraw-Hill Companies, Inc., New York, New York, pp 4-16, (1996)

62. *LightPipes for Matlab, Beam Propagation Toolbox Manual*, version 1.3, OKO Technologies, The Netherlands, (2001)

63. MathWorld.Wolfram.Com, Wolfram Research, "Fast Fourier Transform", http://mathworld.wolfram.com/FFT.html, (2004).

64. Samarskii, A. A., E. S. Nikolaev, *Numerical Methods for Grid Equations*, Volume 1, Direct methods, Birkhauser, Verlag, pp 61-65, (1989)

65. Pedrotti, Frank L., and Leno S. Pedrotti, *Introduction to Optics*, Second Edition, Prentice Hall, Upper Saddle River, New Jersey, pp31-34, (1993).

66. "xcorr", Matlab onscreen Help Navigator.

67. Dimotakis, P.E., H.J Catrakis, and D.C. Fourguette, "Flow Structure and Optical Beam Propagation in High-Reynolds-Number Gas-Phase Shear Layers and Jets", *Fluid Mechanics*, v 433, pp 105-134, 2001.

68. "Prediction and Control of Aero Optic Aberrations", Final Report of Joint AFRL/DAGSI Research Project Number SN-OSU-01-06, 2004.