Polarization Dependence in Nonlinear Fiber Optics: Modeling and Experiment

by

Afrouz Azari
University of Maryland at College Park

Ross Pleban
North Carolina State University

Reza Salem
Graduate Student

Dr. Thomas E. Murphy
Faculty Advisor

Maryland Engineering Research Internship Teams 2006
Department of Electrical and Computer Engineering
University of Maryland at College Park

Acknowledgements

This project was part of the MERIT 2006 program supported by the University of Maryland and the National Science Foundation.

We would like to give our special thanks to Dr. Thomas Murphy, for his immense help and encouragement throughout the program. His support was a great inspiration towards the completion of this project. We would also like to thank Reza Salem, doctoral candidate, for assisting us throughout the summer.

Contents

Abstract

Most currently-used methods for modeling nonlinear propagation in fiber optic do not account for the evolution of the polarization state. A software model has been developed that accounts for the variations in polarization via a numerical approximation and predicts the behavior of a system with different input polarization states. The model uses two methods, an approximation and an exact equation, which have each been derived from the nonlinear Schrödinger equation. The approximation closely agrees with experimental data in which the polarization state frequently changes compared to the length of the fiber. The exact method is free of this constraint but takes longer to simulate. The software tool developed for this project will help to better understand experimental observations of polarization dependence, and will help in designing nonlinear optical switches that are insensitive to polarization.

Introduction

Despite other software models that focus only on specific linear polarization of signal, software developed in this research takes account of most general cases of polarization while giving efficient solution to nonlinear property of fiber optic. This software is able to simulate polarization dependence of different optical fiber effects.

Polarization Dependence Loss

In optical communication it is critical be aware of power loss inside a fiber. Measuring power loss determines how much signal has been attenuated during transmission. If L is length of fiber and P is amount of initial power then power loss can be calculated from equation below:

$$P_T = Pe^{-\alpha L} \tag{1}$$

Where alpha is attenuation constant and its unit is in dB/Km. The relationship between loss and signal power also can be explained as following

$$\alpha_{dB} = -\frac{10}{L}\log(\frac{P_T}{P}) = 4.343\alpha \tag{2}$$

In some cases, fiber loss depends upon polarization state of signal. As a result, any changes in polarization states will change the signal power along the fiber. In a same manner, based on optical component, loss can depend on wavelength. The software model perfectly simulates both polarization- and wavelength-dependent loss.

Group Velocity Dispersion (GVD)

Before getting into the subject of group velocity dispersion, it is convenient to talk about basic concepts of group velocity and phase velocity. Phase velocity is the velocity at which phase of one wave propagates; for example, $V = \frac{c}{n}$ is phase velocity of wave, where c represents speed of light in a vacuum and n is refractive index. On the other hand, group velocity is the rate at which the amplitude of group of propagating wave changes. Simply, phase velocity is speed at which the phase fronts propagate and group velocity is speed of the envelope (wave packet). The relation between phase velocity and group velocity (v_g) can be expressed as followed

$$v_g = c(n - \lambda \frac{dn}{d\lambda})^{-1} \tag{3}$$

Generally speaking, approximation to linear propagation in optical fiber and dispersion can be explained by Taylor expansion:

$$\beta(\omega) = \beta_0 + \beta_1(\omega - \omega_0) + \frac{1}{2}\beta_2(\omega - \omega_0)^2 \tag{4}$$

Where Beta is Propagation constant and is related to refractive index of fiber. The equation for linear propagation is as follow

$$\frac{dA}{dz} = -j\beta_0 A - \beta_1 \frac{dA}{dt} + \frac{j}{2}\beta_2 \frac{d^2 A}{dt^2} \tag{5}$$

Where β_0 is the propagation constant, which is related to the phase velocity, whereas β_1 and β_2 are constant related to group velocity and dispersion respectively. By using the formula above, it can be seen that in process of linear propagation, while wave envelop is propagating along fiber dispersion parameter, β_2, will cause the signal to spread gradually, which is referred to as Group Velocity Dispersion (GVD). The group velocity constant, β_1, can depend on the polarization states, which leads to an effect called Polarization Mode Dispersion (PMD).

Polarization Mode Dispersion (PMD)

As mentioned before, in study of optical fiber, it is desirable to assume that fiber is perfect (polarization-maintaining fiber). However in reality, fiber contains impurities due to the non-ideal asymmetries in the core, mechanical stress such as bending or twisting and environmental temperature. These effects, though in small magnitude, can change the polarization of the wave inside the fiber.

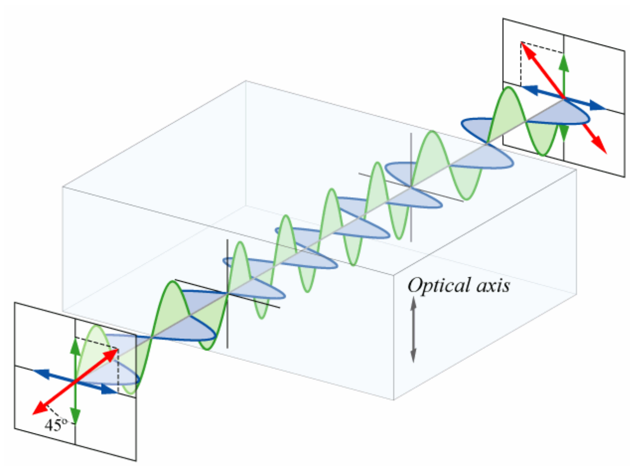

Figure 1: In a birefringent optical fiber, a polarized wave resolves into two orthogonal states that travel with the different speed

Under such condition, propagation constant (Beta) of each polarization states will be different and as a consequence of this modal birefringence, the polarization state will periodically change inside the fiber over a distance called the beat length:

$$L_B = \frac{2\pi}{|\beta_x - \beta_y|} \tag{6}$$

Because each axis has its own propagation constant, their group velocity will be different (categorized to slow and fast axis based on their speed). The process in which the difference between speeds of states will cause the signal broadening and time delay is called Polarization Mode Dispersion. So, polarization dependence of birefringence is called PMD, while wavelength dependence of birefringence is Chromatic Dispersion. In case of Chromatic Dispersion speed of propagation will vary with wavelength.

Nonlinear Characteristics of Fiber Optics

Nonlinearity in optical fiber happens when pulse intensity leads to modulation of refractive index. This phenomenon is called Kerr Effect, which is composed of self-phase and cross-phase modulation.

Self-Phase and Cross-Phase modulation

In case of self-phase modulation, the propagating signal will modulate itself. This effect happens especially when pulse intensity is short enough, which will cause changes in the refractive index. This deviation will cause varying speeds of propagation creating a phase shift.

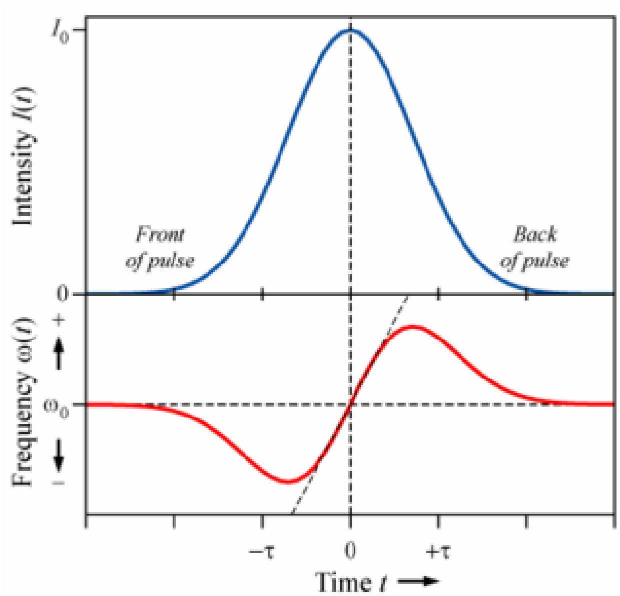

Figure 2: Under effect of self phase modulation, propagating pulse (top) experiences a phase shift (bottom).

Similar to self-phase modulation, cross-phase modulation occurs when more than one signal is propagating inside the fiber. In addition to changing their own speed, a signal will affect the speed of other signals on the optical fiber that are propagating close to them and cause phase modulation.

Method

The evolution of pulse propagation in optical fiber is commonly split into linear and nonlinear equations. Considering only the linear effects it can be modeled by Equation (7). In the equation Â represents the Fourier transform of the optical signal. The pulse shape at any point along the fiber can be obtained from the solution, which is represented by Equation (8). Thus, the linear effects are only seen in the phase of the spectrum or the magnitude of the time domain.

$$\frac{\partial \hat{A}}{\partial z} = \left[-\frac{\alpha(\omega)}{2} - j\beta(\omega) \right] \hat{A} \tag{7}$$

$$\hat{A}(z,\omega) = \hat{A}(z=0,\omega) e^{\left[-\frac{\alpha(\omega)}{2} - j\beta(w) \right] z} \tag{8}$$

Considering only the nonlinear terms pulse propagation results in Equation (9). Gamma, the nonlinearity coefficient, is described in Equation (10). The solution (Eq. 11) to the nonlinear term is obtained in the time domain.

$$\frac{\partial A}{\partial z} = -j\gamma |A|^2 A \tag{9}$$

$$\gamma = \frac{2\pi n_2}{\lambda A_{eff}} \tag{10}$$

$$A(z,t) = A(z=0,t) e^{-j\gamma |A|^2 z} \tag{11}$$

To model the Nonlinear Schrödinger Equation (NLS)

$$\frac{dA}{dz} = -\frac{\alpha}{2}A + \frac{j}{2}\beta_2\frac{d^2A}{dt^2} - j\gamma|A|^2 A \tag{12}$$

we used the Split-step Fourier method. This method divides the optical fiber into many small chunks of equal size. Furthermore each chunk is split in half and a zero length slice is inserted in-between the halves. Each half represents one half of the linear effects of the chunk, and the slice represents the nonlinear effects. Thus, the linear halves and the nonlinear slice can be solved separately.

We implemented the Symmetrized Split-Step Method, which iterates over the nonlinear slice and the second linear half till it converges below a tolerance. This method prevents the grouping of the nonlinearity and has an error on the magnitude of dz^3 compared to the Conventional Split-Step Method error of dz^2.

The NLS equation for modeling pulse propagation assumes light is linearly polarized to an axis or scalar as we like to call it. To take into account the polarization of light we started with the coupled-mode NLS equations that appear in Agrawal's Nonlinear Fiber Optics. These equations are

$$\frac{\partial A}{\partial z} + \beta_{1x}\frac{\partial A_x}{\partial t} + \frac{i\beta_2}{2}\frac{\partial^2 A_x}{\partial t^2} + \frac{\alpha}{2}A_x = i\gamma\left(\left|A_x\right|^2 + \frac{2}{3}\left|A_y\right|^2\right)A_x + \frac{j\gamma}{3}A_x^* A_y^2 e^{-2i\Delta\beta z} \tag{13}$$

$$\frac{\partial A}{\partial z} + \beta_{1y}\frac{\partial A_y}{\partial t} + \frac{i\beta_2}{2}\frac{\partial^2 A_y}{\partial t^2} + \frac{\alpha}{2}A_y = i\gamma\left(\left|A_y\right|^2 + \frac{2}{3}\left|A_x\right|^2\right)A_y + \frac{j\gamma}{3}A_y^* A_x^2 e^{-2i\Delta\beta z} \tag{14}$$

where $\Delta\beta$ is $\beta_{0X} - \beta_{0Y}$. Instead of $\Delta\beta$, we use an effective phase velocity that is the difference between the actual phase velocity of the axis and the average phase velocity of both axes. The same procedure was done for the group velocities.

The last term in Equations (13, 14) can be assumed to be zero when the fiber length is much greater than the beat length or the polarization state frequently changes compared to the length of the fiber. This assumption greatly simplifies the calculations to model pulse propagation. This approximation does not need to be made, if a rotation is made to a circular coordinate system.

Elliptical Birefringence

All the previous propagation equations have only considered linear birefringent optical fibers. However, this is not always the case; the eigenstates of the fiber could be elliptically polarized. Two variables, χ and ψ, are needed to entirely describe the birefringence polarization. χ is the degree of ellipticity and ψ is the angular orientation of the ellipse relative to the x-axis. A visual representation is shown in Figure 3 below:

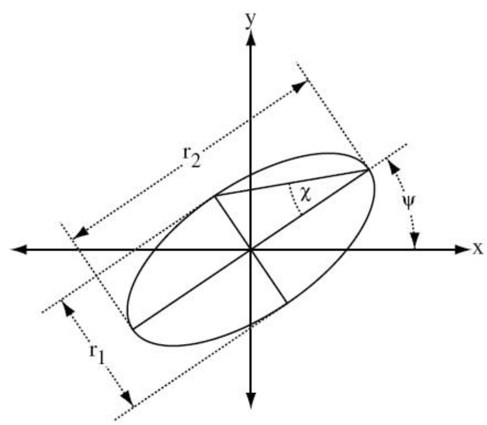

Figure 3: Elliptical Birefringence Parameters

Approximate Method Using Eigenstates of the Fiber

One of the two available methods of modeling the NLS equations available is the approximation of Equations (13, 14) using the eigenstates of the fiber. This method is referred to as "elliptical" throughout the remainder of paper. In our model we assume that the input vectors are the electric fields corresponding to the original x-y axes; thus, we must rotate to orthogonal eigenstates, u_a and u_b, of the birefringent fiber using

$$\begin{bmatrix} u_a \\ u_b \end{bmatrix} = \begin{bmatrix} \cos\psi\cos\chi - j\sin\psi\sin\chi & \sin\psi\cos\chi + j\cos\psi\sin\chi \\ -\sin\psi\cos\chi + j\cos\psi\sin\chi & \cos\psi\cos\chi + j\sin\psi\sin\chi \end{bmatrix} \times \begin{bmatrix} u_x \\ u_y \end{bmatrix} \tag{15}$$

We will now consider the linear and nonlinear propagation equations separately due to its application to the Split-Step Fourier Method. The linear propagation including the rapid oscillations of polarization approximation in terms of the new coordinate system is

$$\begin{bmatrix} \hat{u}_a(z,\omega) \\ \hat{u}_b(z,\omega) \end{bmatrix} = \begin{bmatrix} h_a & 0 \\ 0 & h_b \end{bmatrix} \times \begin{bmatrix} \hat{u}_a(0,\omega) \\ \hat{u}_b(0,\omega) \end{bmatrix} \tag{16}$$

where h_a and h_b are

$$h_a = \exp\left\{\left[-\frac{\alpha_a(\omega)}{2} - j\beta_a(\omega)\right]z\right\} \tag{17}$$

$$h_b = \exp\left\{\left[-\frac{\alpha_b(\omega)}{2} - j\beta_b(\omega)\right]z\right\} \tag{18}$$

The nonlinear propagation can be calculated by:

$$\frac{\partial u_a}{\partial z} = -j\frac{\gamma}{3}\left[(2+\cos^2 2\chi)|u_a|^2 + (2+2\sin^2 2\chi)|u_b|^2\right]u_a + \ldots \tag{19}$$

$$\frac{\partial u_b}{\partial z} = -j\frac{\gamma}{3}\left[(2+\cos^2 2\chi)|u_b|^2 + (2+2\sin^2 2\chi)|u_a|^2\right]u_b + \ldots \tag{20}$$

where the (…) terms are additional nonlinear terms that average to zero in the case of high birefringence. After the Split-Step iterations are done, the model must rotate back to the original x-y axes using

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = \begin{bmatrix} \cos\psi\cos\chi + j\sin\psi\sin\chi & -\sin\psi\cos\chi - j\cos\psi\sin\chi \\ \sin\psi\cos\chi - j\cos\psi\sin\chi & \cos\psi\cos\chi - j\sin\psi\sin\chi \end{bmatrix} \times \begin{bmatrix} u_a \\ u_b \end{bmatrix} \tag{21}$$

Exact Method Using Circular Basis

The second method available in the model is the circular basis of Equations (13, 14). The input orientation is the same as the approximate method (x-y axes). The rotation to the circular coordinate system is made by:

$$\begin{bmatrix} u_+ \\ u_- \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & j \\ j & 1 \end{bmatrix} \times \begin{bmatrix} u_x \\ u_y \end{bmatrix} \qquad (22)$$

where u_+ represents the left-hand circularly-polarized component and u_- represents the right-hand circularly-polarized component. This rotation is the special case of the rotation in Eq. (15) when $\chi=\pi/4$ and $\psi=0$. The linear propagation in the circular basis including elliptical birefringence follows:

$$\begin{bmatrix} \hat{u}_+(z,\omega) \\ \hat{u}_-(z,\omega) \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \times \begin{bmatrix} \hat{u}_+(0,\omega) \\ \hat{u}_-(0,\omega) \end{bmatrix} \qquad (23)$$

where h_{mn} are define as:

$$h_{11} = \frac{1}{2}\left[(1+\sin 2\chi)h_a + (1-\sin 2\chi)h_b\right]$$

$$h_{12} = -\frac{j}{2}e^{+j2\psi}\cos 2\chi(h_a - h_b)$$

$$h_{21} = +\frac{j}{2}e^{-j2\psi}\cos 2\chi(h_a - h_b)$$

$$h_{22} = \frac{1}{2}\left[(1-\sin 2\chi)h_a + (1+\sin 2\chi)h_b\right]$$

The nonlinear propagation equations in the circular basis are

$$\frac{\partial u_+}{\partial z} = -j\frac{2\gamma}{3}\left[|u_+|^2 + 2|u_-|^2\right]u_+ \tag{24}$$

$$\frac{\partial u_-}{\partial z} = -j\frac{2\gamma}{3}\left[|u_-|^2 + 2|u_+|^2\right]u_- \tag{25}$$

After the Split-Step iterations are finished a rotation must be made from the circular basis to the x-y axes using

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -j \\ -j & 1 \end{bmatrix} \times \begin{bmatrix} u_+ \\ u_- \end{bmatrix} \tag{26}$$

MATLAB Model

The model was written as a MATLAB function that included both methods, elliptical and circular, of solving the NLS equation. The function encompasses all rotations and

Input Parameters:

`u0x, u0y`	`Starting field amplitude`
`dt`	`Time step`
`dz`	`Propagation step size`
`nz`	`Number of steps to take (i.e. ztotal = dz*nz`
`alphaa, alphab`	`Power loss coefficients for the two fiber eigenstates`
`betapa, betapb`	`Dispersion polynomial coefs, [beta_0 ... beta_m] for the two eigenstates`
`gamma`	`Nonlinearity coefficient`
`pol = [chi,psi]`	`Polarization state of the primary fiber eigenstate (default = [0,0])`
`method`	`Which method to use, either ’circular’ or ’elliptical’ (default = ’elliptical’)`
`maxiter`	`Max number of iterations per step (default = 4)`
`tol`	`Convergence tolerance (default = 1e-5)`

Output Parameters:

`u1x, u1y`	`Output field amplitudes`

The starting and output electric fields are vectors of any length that describe the slowly varying waveforms. The input and output are in the x-y basis and can be polarized to any state.

The `pol` parameter describes the birefringence of the fiber via χ and ψ, which correspond to the eigenstates of the fiber. See Figure 3 for a visual description. $\text{Alpha}_{a,b}$ and $\text{betap}_{a,b}$ represent the power attenuation constant and the propagation constant respectively. These parameters are aligned with the eigenstates of the fiber. They can be declared as a vector of Taylor Series coefficients or as a vector the same length as `u0x`. If their length is less than u0x, then the model will automatically calculate the corresponding approximation of α(ω) and β(ω). If $alpha_{a,b}$ and $betap_{a,b}$ are the same length as u0x, then they will be directly interpreted as α(ω) and β(ω).

The last four input parameters (pol, method, maxiter, and tol) are all optional. To specify one of the last four parameters, the optional parameters above must also be specified. For example, to declare maxiter you must declare pol and method. If pol is not specified, then it will be assumed the birefringence is aligned with the x-y axes. The default method is elliptical.

The dimensions of the input and output quantities are arbitrary, as long as they are self consistent. For example, if $|u0x|^2$ has dimensions of Watts and dz has dimensions of meters, then the nonlinearity parameter, gamma, should be specified in $W^{-1}m^{-1}$. Similarly, if dt is given in picoseconds, and dz is given in meters, then the dispersion polynomial $betap^{(n)}$ should have dimensions of $ps^{(n-1)}/m$.

C Model

The model was also developed in the C programming language to increase the speed of simulation. The C version has the same exact input and output parameters as the MATLAB version. C is at a lower level of abstraction than MATLAB so it provides control of data storage and movement. MATLAB stores matrices columnwise whereas C stores matrices horizontalwise. The storage method is important, because the model requires heavy fft and ifft calculations. These calculations are done using the FFTW library, which stores matrices horizontalwise. Therefore, MATLAB must continuously convert its matrices back and forth between the two storage conventions. Implementing the C version will only require this conversion for the initial function call and for the return vectors.

FFTW must make a plan for each fft and ifft with different lengths, input vectors or output vectors. This plan is a run-time analysis of the current computer's architecture and is an attempt to optimize FTTW's performance. The C version of the model has the ability to save and load the necessary plans.

Results

We verified the accuracy of our model by testing it against an earlier, scalar version of the code and known soliton solutions. We also compared our model to experimental data of cross-phase modulation in Bismuth-Oxide-Based highly nonlinear fiber. Additionally we ran benchmark tests to analyze the run-time speeds of the different versions and methods.

Linear Verification

To ensure that the model was correct in only linear simulations we compared our output to the output of the scalar version, which can only handle linearly polarized inputs. To mimic the scalar model we set the characteristics of the eigenstates of the optical fiber equal to one another; thus the input polarization state should not affect the output. The parameters of the simulation were:

$$\gamma = 0$$
$$\beta_{0x} = \beta_{0y} = 0$$
$$\beta_{1x} = \beta_{1y} = 0$$
$$\beta_{2x} = \beta_{2y} = -1$$
$$L = \frac{\pi}{2}$$
$$u_x(0,t) = A_x \times \mathrm{sech}(t)$$
$$u_y(0,t) = A_y \times \mathrm{sech}(t)$$

where A_X and A_Y are their respective element from the Jones Vector that describes polarization. Multiple polarization states of linear, circular, and elliptical were tested along with each method, elliptical and circular. This simulation was done with a single step, nz = 1, since there is no nonlinearity. The result of this simulation for both the scalar version and our model for any polarization state or method was

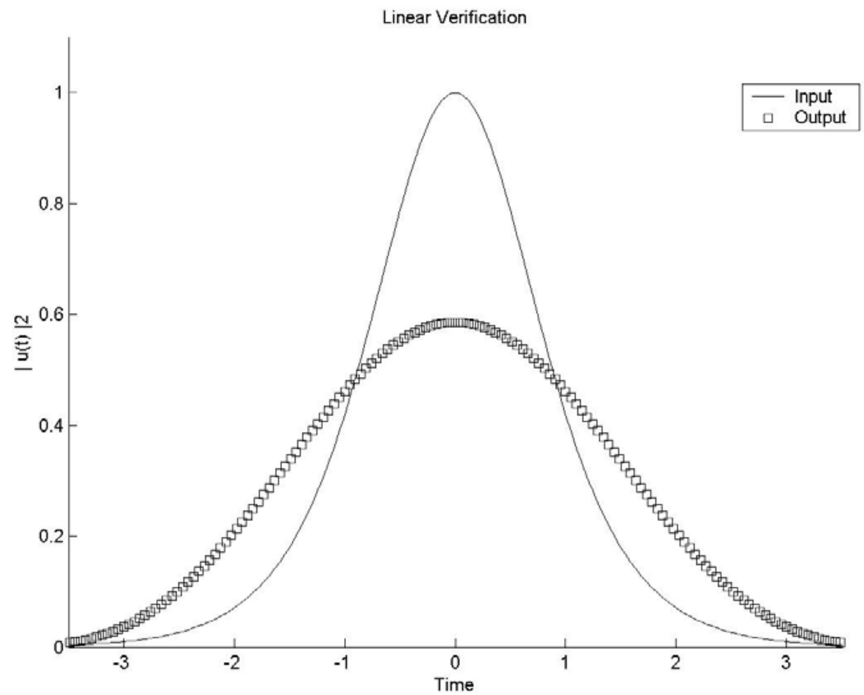

Figure 4: Linear verification for all polarization states and methods

Nonlinear Verification

To confirm the accuracy of the nonlinear workings of our model, we tested a known soliton solution. A soliton is a special case when the dispersion and nonlinearity cancel one another resulting in the output equal to the input. The parameters were

$$\gamma = 1$$
$$\beta_{0x} = \beta_{0y} = 0$$
$$\beta_{1x} = \beta_{1y} = 0$$
$$\beta_{2x} = \beta_{2y} = -1$$
$$L = \frac{\pi}{2}$$
$$u_x(0,t) = A_x \times \text{sech}(t)$$
$$u_y(0,t) = A_y \times \text{sech}(t)$$

This test case required multiple steps in order for the steps to converge. The output for all polarization states using the circular method and linear polarization to one axis using the elliptical method was

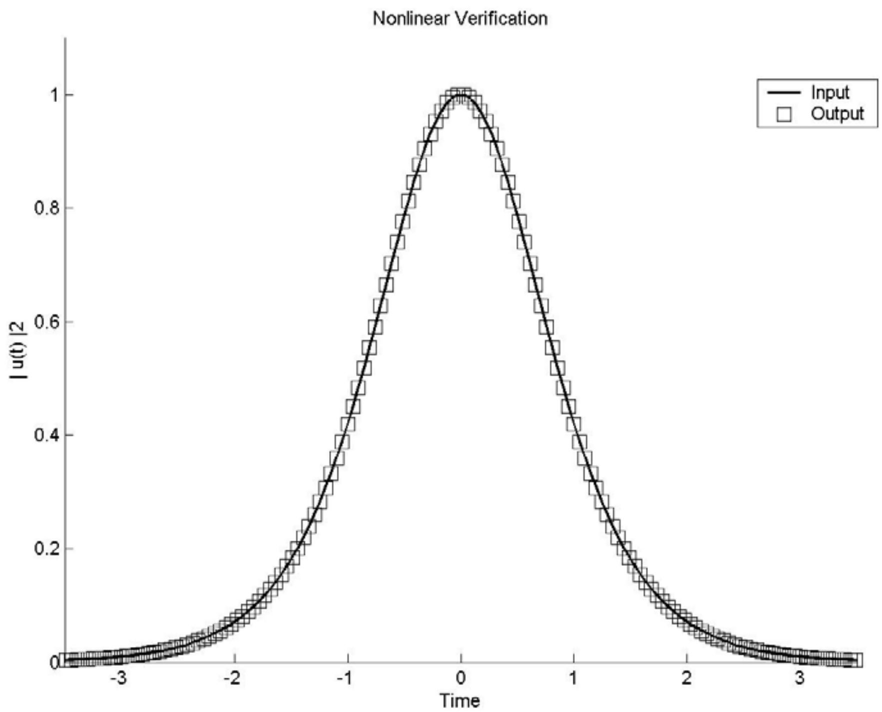

Figure 5: Nonlinear verification for all polarizations for the circular method and linearly polarized to one axis for the elliptical method

It is significant to note that to produce the soliton solution in the case of circularly polarized using the circular method we set gamma equal to three halves. This is expected since its nonlinearity factor is weaker by two thirds.

The elliptical method did not produce the soliton solution for polarization states that were not linearly polarized to the x- or y-axis. The output for these cases was

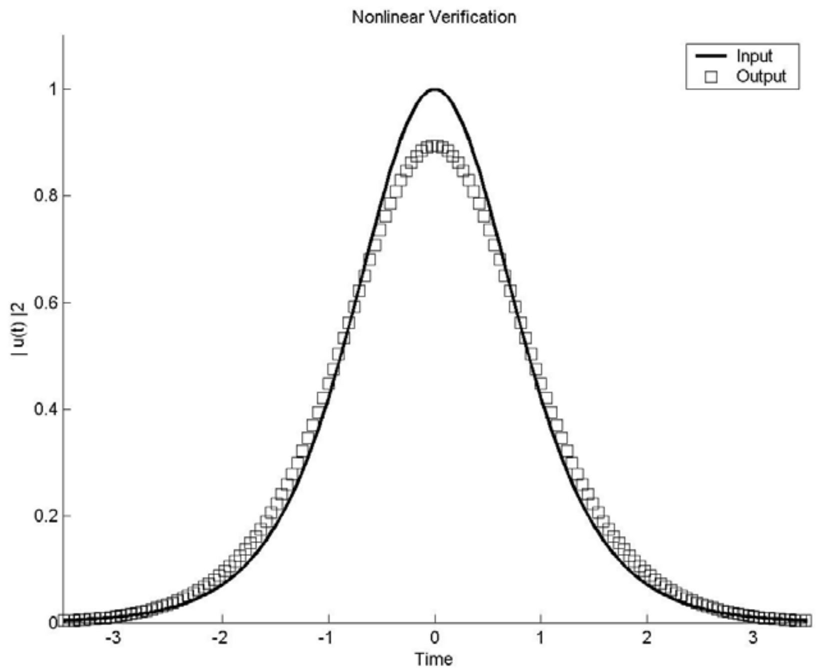

Figure 6: Nonlinear verification of polarizations not aligned with an axis using the elliptical method

The output pulse was broader than the input; thus, not a correct soliton solution.

Experimental Comparison

We compared our model to experimental data of cross-phase modulation in Bismuth-Oxide-Based highly nonlinear fiber. The experimental data was from earlier experiments conducted at the University of Maryland. The experiment involved optically modulating a data signal by a clock signal. The output of interest is the filtered data signal and it is what we will examine. The parameters of our model were setup to exactly match those used in the experiment. The following is the filtered output comparison of the experimental data and our model using the circular method:

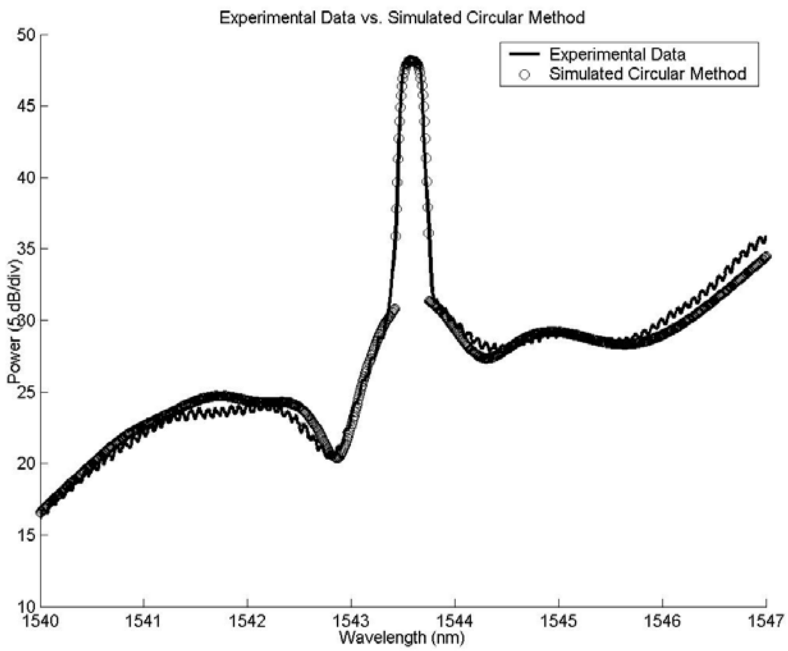

Figure 7: Experimental data versus simulated circular method

The comparison of the filtered output of the experimental data and our model using the elliptical method is

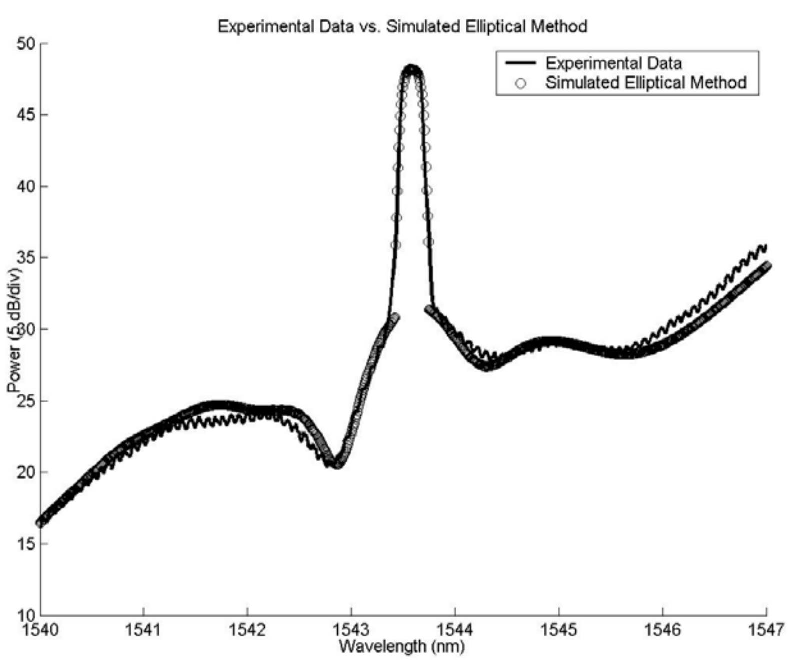

Figure 8: Experimental data versus simulated elliptical method

Benchmark Tests

Run-time benchmark tests were ran to measure the speed difference between the different versions and methods. The test setup was one signal from the cross-phase modulation experiment. The time measurement was from the CPU time before and after the simulation according to MATLAB. The benchmark graph in which the number of steps was two hundred for all versions and methods is

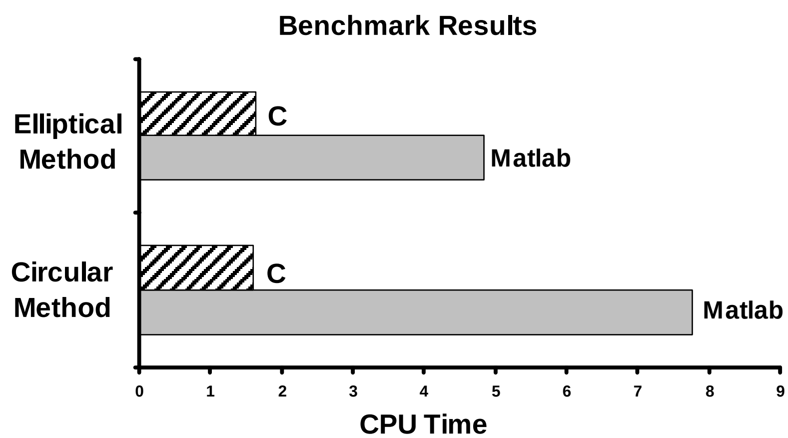

Figure 9: Benchmark results with equal number of steps between methods

Additional tests were run in which the number of steps used in the circular method was two hundred and in the elliptical method was sixty. The output between the two methods was no different than when the same number of steps was used. The results of this test are

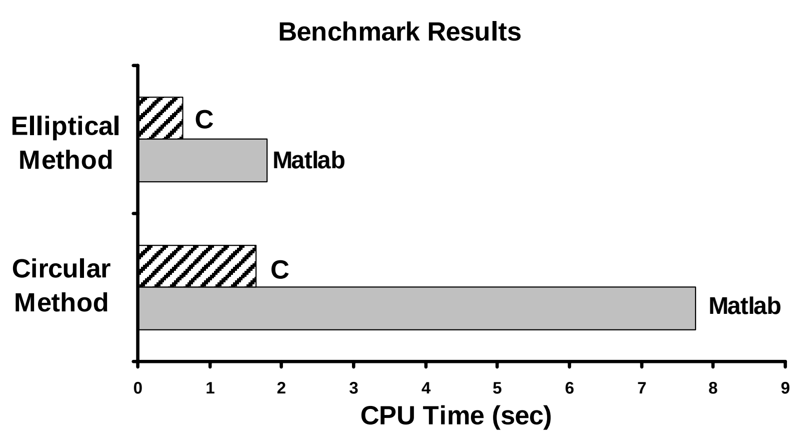

Figure 10: Benchmark results with fewer steps in the elliptical method

Discussion

The linear verification tests show that both methods in our model exactly match the scalar model. It is also shown that if there is no nonlinearity, then the entire fiber can be simulated in one step. This is expected since we are using a known solution for the linear part and the steps are only needed to approximate the nonlinearity.

The nonlinear verification tests did not produce the same results for the circular and elliptical methods. The circular method generates the precise soliton solution for any input polarization. The elliptical method will only correctly simulate the test when the input is polarized to either the x- or y-axis. If the input is any other polarization state, then an error is introduced. This error is due to the approximation of Equations (13, 14) in which the last term was ignored. The basis of this approximation is that the rapid polarization oscillations will cause the term to average to zero. In other words if the length of the fiber is much greater than the beat length, then it is a valid approximation. In our soliton test case, the beat length is infinite since there is no birefringence. The fiber length of $\pi/2$ is most definitely smaller than infinity, which explains why in this case the elliptical method should not produce accurate results.

The comparison of the cross-phase modulation experimental data and our model show an excellent match between theory and experiment. This is a particularly thorough test of our model since it employs most of the effects that are included in the model. Both methods, circular and elliptical, closely follow the experimental data. Also the circular and elliptical methods are almost exact replicas of each other proving that the elliptical approximation is useful and valid.

The benchmark tests prove that our work in coding a C version of the model was not futile. In the tests with equal number of steps the C version runs 65 – 80% faster than the MATLAB version. If we take advantage of the elliptical approximation and reduce the number of steps, then the elliptical version's speed is increased by 62%. Most impressively the elliptical C version is 93% faster than the circular MATLAB version.

Conclusion

We have succeeded in creating a general purpose software model that simulates polarization evolution in nonlinear fiber optics. Our model is versatile and will assist in a wide range of fiber optic research areas. It allows for hundreds of permutations of variables to be simulated in minutes that would take days to reproduce in the lab. In particular we know that it will be helpful in finding polarization insensitive optical switches. Optical switches are needed to remove the delay in converting from light to electricity and back. Polarization insensitive switches will eliminate the need to maintain the polarization throughout the entire fiber.

Our model with be free and publicly available under the GNU Public License. People around the world already use the scalar version of the code, and there is a known demand for the full-vector version. The model will also be useful in learning environments.

References

[1] Govind P. Agrawal. *Nonlinear Fiber Optics*. Academic Press, 2001.

[2] Reza Salem, A. S. Lenihan, G. M. Carter, and T. E. Murphy. 160 Gb/s Polarization-Independent Optical Demultiplexing in 2-m Bismuth-Oxide Fiber. In *IEEE Photon. Technol. Lett.,* **18**(12), 1329 – 1331. 2006.

[3] Thomas E. Murphy. Split Step Fourier Propagation Software. Photonics Research Laboratory. 2006. Available at http://www.photonics.umd.edu/software/ssprop/.

[4] FFTW Home Page. http://www.fftw.org/