A Fast Method in Simulations of Supercontinuum Generation in Photonic Crystal Fibers

Shuang Jin

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of Master of Applied Science at

Concordia University

Montreal, Quebec, Canada

December 2005

Your file Votre référence
ISBN: 0-494-14263-4
Our file Notre référence
ISBN: 0-494-14263-4

Abstract

A Fast Method in Simulations of Supercontinuum Generation in Photonic Crystal Fibers

Shuang Jin

Over the past few years photonic crystal fibers (PCFs) had shown potential for optical supercontinuum (SC) generations that permitted significant progress in various applications such as optical metrology, medical science or telecommunications. Simulation of SC generation played a crucial role in demonstrating the physical mechanisms behind the process of SC generation in PCFs although many experimental studies have shown the novel phenomenon.

In this thesis, general nonlinear Schrödinger equation is solved in very detailed numerical process by using Symmetrized Split-Step Fourier Method (S-SSFM), Higher-Order Symmetrized Split-Step Fourier Method (HO-S-SSFM) and Predictor-Corrector Symmetrized Split-Step Fourier Method (PC-S-SSFM). We simulated the SC generations in the PCF by using HO-S-SSFM and PC-S-SSFM respectively. To our knowledge, this is the first time the PC-S-SSFM is used in simulations of SC generation. Typical physical parameters of a PCF are chosen in our simulations. Based on the same condition, the simulation results of the HO-S-SSFM and the PC-S-SSFM are compared with the results from a published paper, and the comparison proves that our simulations are correct and accurate. Then, the time used by the HO-S-SSFM and the PC-S-SSFM in our simulations are compared at different points of propagation along the PCF and at the end of the PCF with different step sizes. The comparison results show that the PC-S-SSFM can save almost one third time used by the HO-S-SSFM under the same accuracy.

Acknowledgements

I am deeply grateful to my supervisor Dr. **John Xiu pu Zhang** for his initiating this project, and for his research advices, his teaching and inspiration in all phases of this research work. Many thanks and appreciation is due to my family and friends for their encouragement and moral supports.

Table of Contents

List of Figures

List of Acronyms and Symbols

SC	Supercontinuum
PCF	Photonic crystal fiber
DWDM	Dense wavelength-division multiplexing
SPM	Self-phase modulation
GNLS	General Nonlinear Schrödinger Equation
SSFM	Split-Step Fourier Method
S-SSFM	Symmetrized Split-Step Fourier Method
HO-S-SSFM	Higher Order Symmetrized Split-Step Fourier Method
PC-S-SSFM	Predictor-Corrector Symmetrized Split-Step Fourier Method
SS	Self-steepening
SRS	Stimulated Raman scattering
FWM	Four-wave mixing
SEM	Scanning electron micrograph
OSA	Optical spectrum analyzer
FWHM	Full width at half maximum
γ	Nonlinear coefficient
n_2	Nonlinear refractive index
A_{eff}	Fiber effective area
ω_0	Center frequency
c	Speed of light in vacuum
β	Propagation constant
α	Fiber loss

h	Step size
z	Replacement of optical pulse propagating
T	Relative time of optical pulse experiencing

Chapter 1

Introduction

A supercontinuum (SC) is a broad spectrum extending beyond all visible colors with the properties of a laser. In other words, a supercontinuum is coherent white light. The first observation of a supercontinuum dates back to 1970, when Alfano and Shapiro focused powerful picosecond pulses into a glass sample [1]. Supercontinuum generation was achieved in a conventional single-mode optical fiber in 1987 [2], [3]. Several fiber designs have been proposed to enhance the generated bandwidth. Supercontinuum generation in a photonic crystal fiber (PCF) was demonstrated in 1999 by Ranka *et al.* [4]. With newly invented microstructure photonic crystal fibers, the spectral broadening is so pronounced that the resulting frequency comb can span more than an optical octave.

A supercontinuum light source finds numerous novel applications in the field of telecommunication [5], optical metrology [6] and medical science. The interest in applying supercontinuum sources in dense wavelength-division multiplexing (DWDM) transmission is increasing. By slicing the broad spectrum of the supercontinuum into hundreds of channels, and utilizing an optical time domain multiplexing technique for each channel, transmission bandwidths of terahertz can be achieved [7]. Also the use of a supercontinuum in single-shot characterization of fiber-optic components has been demonstrated [8]. The continuum consists of millions of peaks equally spaced by the repetition rate of the laser [9]. Indeed, the usage of a supercontinuum generated in a

photonic crystal fiber in the creation of a stabilized frequency comb provides convenient means to link optical frequency standards together. In addition, the relation between the repetition rate of the pulses and the comb spacing has provided a link between optical and microwave frequencies for the first time. This enables comparison of the performance of Cesium atomic clocks with stabilized lasers.

The physics behind the process of supercontinuum generation in photonic crystal fibers has been studied since the results of Ranka *et al.*, and several attempts have been made to explain the generated broad bandwidth [10], [11], [12]. The dominant nonlinear effects responsible for the continuum generation are expected to be self-phase modulation (SPM), self-steepening (SS), intrapulse Raman scattering and four-wave mixing (FWM). To acquire a better understanding of the physical mechanisms of the process, and to study the effects of the supercontinuum generation in a photonic crystal fiber, simulations of supercontinuum generation in photonic crystal fibers become more and more significant in this area.

The objective of this thesis is to numerically solve a generalized scalar propagation equation, which is derived from General Nonlinear Schrödinger Equation (GNLS) and to find a faster approach to present the simulated results which are satisfied in studying the supercontinuum generation in photonic crystal fibers. In Chapter 2 of this thesis, basic concepts of supercontinuum and Photonic crystal fibers are described with some figures. In Chapter 3, a generalized scalar propagation equation that governs propagation of optical pulses in PCF is presented. Solving the GNLS by using Symmetrized Split-Step Fourier Method (S-SSFM), Higher Order Symmetrized Split-Step Fourier Method (HO-S-SSFM) and Predictor-Corrector Symmetrized Split-Step

Fourier Method (PC-S-SSFM) are discussed in details. In Chapter 4, the simulations by using the two different ways mentioned above are given and the results of the simulations are compared with the results of a published paper. Also the Higher Order Split-Step Fourier Method and the Predictor-Corrector Split-Step Method are compared by using time efficiency. Finally, conclusion is found in Chapter 5 of this thesis that PC-S-SSFM is an effective method of SC generation in photonic crystal fibers, and it is a new technique to accelerate S-SSFM.

Chapter 2

Supercontinuum and Photonic Crystal Fibers

2.1 Supercontinuum

The interaction of intense pulses with a nonlinear medium can lead to considerable broadening of the pulse spectrum. The resulting spectrum can exceed several hundreds of nanometers, which is commonly referred to as supercontinuum (SC). The properties of the SC depend critically on the input pulse parameters and on the medium in which it is generated. When the SC covers the whole spectral visible region, it appears as white light as shown in Figure 1[13].

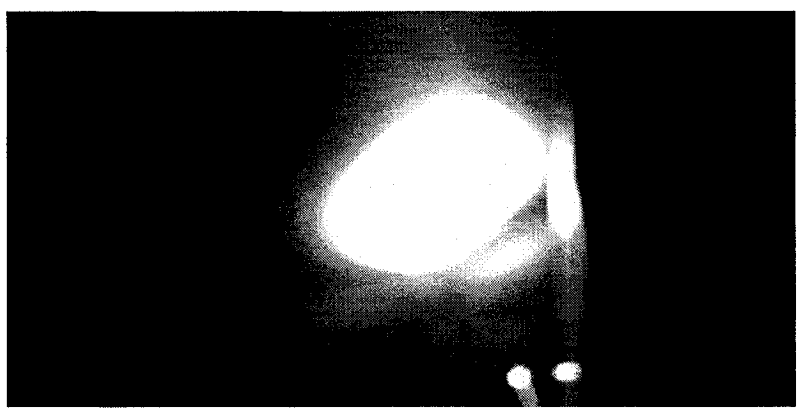

Figure 1. Image of a supercontinuum generated in a photonic crystal fiber and recorded with a digital camera.

The supercontinuum has shown potential for many practical applications and permits significant progress in various domains such as telecommunications, optical metrology and medical science.

2.2 Photonic Crystal Fibers

Photonic crystal fibers were first demonstrated in 1970 [14]. They became commercially available in the late nineties by the development of new fiber manufacturing technologies. Photonic crystal fibers exhibit a novel structure compared with that of conventional optical fibers. Conventional optical fibers consist of a cladding made of silica, and a doped silica core of which the refractive index is slightly higher than that of the cladding. According to the principle of total internal reflection, difference in the refractive index between the core and the cladding allows to guide light within the core of the fiber. In a PCF, the cladding is formed by introducing a periodic pattern of air-holes around a silica core. The air-holes lower the effective refractive index of the cladding compared to the solid silica core, therefore light coupled into the fiber is guided within the core due to reflection from the glass-air interface. Control of the air-holes of photonic crystal fibers permits to tailor the dispersion profile and the efficiency of nonlinear effects [15].

Photonic crystal fibers are fabricated by stacking small glass capillary tubes around one solid glass tube which will form the core of the fiber as shown in Figure 2 [13].

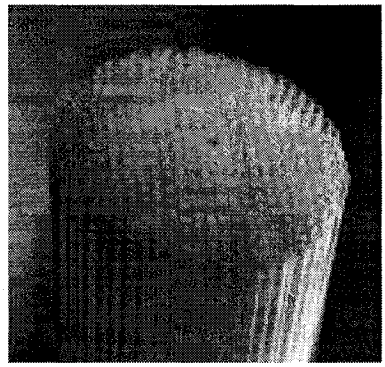

Figure 2. Preform of a photonic crystal fiber.

The optical properties of photonic crystal fibers differ from those of conventional fibers. The index difference between the core and the cladding and the effective core area can easily be tailored to obtain properties for the intended application. Varying the air-hole size, arrangement and symmetry allows for tailoring the group-delay and the dispersion properties of the fiber. New technique used to fabricate photonic crystal fibers permits to reduce the core diameter down to 1.0 μm (see Figure 3 [13]), thus increasing considerably the efficiency of nonlinear processes since such processes are approximately inversely proportional to the area of the fiber core. This makes photonic crystal fibers particularly suitable for supercontinuum generation where high nonlinearities are required. These nonlinearities include self-phase modulation (SPM), self-steepening (SS), stimulated Raman scattering (SRS) and four-wave mixing (FWM)[13].

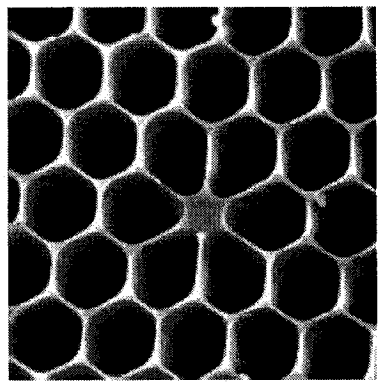

Figure 3. Scanning electron micrograph (SEM) of a small core PCF.

Chapter 3

Higher-Order Symmetrized Split-Step Fourier Method and Predictor-Corrector Symmetrized Split-Step Fourier Method

3.1 Numerical Model

The supercontinuum generation was modelled by using a generalized scalar propagation equation suitable for studying broad-band pulse evolution in optical fibers [16], [17]

$$\frac{\partial A(z,T)}{\partial z} = -\left(\sum_{m=2} \beta_m \frac{i^{m-1}}{m!} \frac{\partial^m}{\partial T^m} + \frac{\alpha}{2}\right) A(z,T)$$

$$+ i\gamma \left(1 + \frac{i}{\omega_0} \frac{\partial}{\partial T}\right) \left(A(z,T) \int_{-\infty}^{\infty} R(t') \left|A(z,T-t')\right|^2 dt'\right) \quad (1)$$

Here, $A = A(z,T)$ is the electric field envelope, the β_m's are the usual dispersion coefficients at center frequency ω_0 , $\gamma = n_2 \omega_0/(c A_{\text{eff}})$ is the nonlinear coefficient with n_2 the nonlinear refractive index , c the speed of light in vacuum and A_{eff} the fiber effective area, and α is the fiber loss. The very right hand side of Equation (1) models self-phase

modulation, self-steepening the optical shock formation, and stimulated Raman scattering. The response function $R(t)$ can be written as [17], [18], [19].

$$R(t) = (1 - f_R)\delta(t) + f_R h_R(t). \tag{2}$$

it includes both instantaneous electronic and delayed Raman contributions. For $h_R(t)$, experimental attempts have been made to determine an approximate analytic form of the Raman response of function [17].

$$h_R(t) = \frac{\tau_1^{\,2} + \tau_2^{\,2}}{\tau_1 \tau_2^{\,2}} \exp(-t/\tau_2)\sin(t/\tau_1).$$

The parameters τ_1 and τ_2 are two adjustable parameters and are chosen to provide a good fit to the actual Raman-gain spectrum. Their appropriate values are τ_1 =12.2 fs and τ_2 = 32 fs [17]. By using the known numerical value of peak Raman gain f_R is estimated to be about 0.18 [17], [18], [19]. Equation (1) together with the response function $R(t)$ given by Equation (2) governs evolution of the pulses in optical fibers. When $R(t)$ is replaced by Equation (2), optical propagation Equation (1) can be written as Equation (3).

$$\frac{\partial A(z,T)}{\partial z} = -\left(\sum_{m=2} \beta_m \frac{i^{m-1}}{m!}\frac{\partial^m}{\partial T^m} + \frac{\alpha}{2}\right)A(z,T)$$

$$+ i\gamma\left(1 + \frac{i}{\omega_0}\frac{\partial}{\partial T}\right)\left(A(z,T)\int_{-\infty}^{\infty}[(1-f_R)\delta(t') + f_R h_R(t')]\left|A(z,T-t')\right|^2 dt'\right) \tag{3}$$

Equation (3) is solved as Equation (4)

$$\frac{\partial A(z,T)}{\partial z} = -\left(\sum_{m=2} \beta_m \frac{i^{m-1}}{m!} \frac{\partial^m}{\partial T^m} + \frac{\alpha}{2}\right) A(z,T)$$

$$+ i\gamma \left(1 + \frac{i}{\omega_0} \frac{\partial}{\partial T}\right) \left(A(z,T)(1-f_R)\left|A(z,T)\right|^2 + f_R A(z,T) h_R(T) * \left|A(z,T)\right|^2\right), \quad (4)$$

where $h_R(T) * \left|A(z,T)\right|^2$ is the convolution of $h_R(T)$ and $\left|A(z,T)\right|^2$.

By calculating $\partial / \partial T$ at the very right side part of Equation (4) and using $A = A(z,T)$ to simplify the notation, Equation (4) becomes Equation (5).

$$\frac{\partial A}{\partial z} = -\left(\sum_{m=2} \beta_m \frac{i^{m-1}}{m!} \frac{\partial^m}{\partial T^m} + \frac{\alpha}{2}\right) A$$

$$+ i\gamma \left\{(1-f_R)\left(A\left|A\right|^2 + \frac{i}{\omega_0} \frac{\partial}{\partial T}(A\left|A\right|^2)\right) + f_R \left(1 + \frac{i}{\omega_0} \frac{\partial}{\partial T}\right)\left(A h_R(T) * \left|A\right|^2\right)\right\} \quad (5)$$

Numerical solutions to the Equation (5) are going to use the Split-Step Fourier Method.

3.2 Split-Step Fourier Method

The GNLS equation (5) is a nonlinear partial differential equation that does not generally lend itself to analytic solutions except for some specific case. A numerical

approach is therefore often necessary for an understanding of the nonlinear effects in optical fibers. Split-Step Fourier Method is an extensive way to solve the pulse-propagation problem in nonlinear dispersive media.

To understand the philosophy behind the Split-Step Fourier Method, it is useful to write Equation (5) formally in the form as Equation (6) [20].

$$\frac{\partial A}{\partial z} = (\hat{D} + \hat{N})A, \tag{6}$$

where $\hat{D}$ is a differential operator that accounts for dispersion and absorption in a linear medium and $\hat{N}$ is a nonlinear operator that governs the effect of fiber nonlinearities on pulse propagation. These operators are given by

$$\hat{D} = -\sum_{m=2} \beta_m \frac{i^{m-1}}{m!} \frac{\partial^m}{\partial T^m} - \frac{\alpha}{2} \tag{7}$$

$$\hat{N} = i\gamma \left\{ (1 - f_R) \left(|A|^2 + \frac{i}{\omega_0} \frac{1}{A} \frac{\partial}{\partial T} (A|A|^2) \right) \right\}$$

$$+ f_R \left[h_R(T) * |A(z,T)|^2 + \frac{i}{\omega_0} \frac{1}{A} \frac{\partial}{\partial T} \left(A h_R(T) * |A(z,T)|^2 \right) \right] \tag{8}$$

In general, dispersion and nonlinearity act together along the length of the fiber. The Split-Step Fourier Method obtains an approximate solution by assuming that in

propagating the optical field over a small distance h, the dispersive and nonlinear effects can be pretended to act independently. More specifically, propagation from z to $z + h$ is carried out in two steps. In the first step, the nonlinearity acts alone, and $\hat{D} = 0$ in Equation (7). In the second step, dispersion acts alone, and $\hat{N} = 0$ in Equation (8). Mathematically,

$$A(z+h,T) \approx \exp(h\hat{D})\exp(h\hat{N})A(z,T) \tag{9}$$

The exponential operator $\exp(h\hat{D})$ can be evaluated in the Fourier domain using the prescription [20]

$$\exp(h\hat{D})A(z,T) \approx F_T^{-1}\exp[h\hat{D}(i\omega)]F_T A(z,T), \tag{10}$$

where F_T denotes the Fourier-transform operation, $\hat{D}(i\omega)$ is obtain from Equation (7) by replacing the differential operator $\partial/\partial T$ by $i\omega$, and ω is the frequency in the Fourier domain.

To estimate the accuracy of the Split-Step Fourier Method, we note that a formally exact solution of Equation (6) is given by

$$A(z+h,T) = \exp[h(\hat{D}+\hat{N})]A(z,T) \tag{11}$$

if $\hat{N}$ is assumed to be z independent. At this point, it is useful to recall the Baker-Hausdorff formula [21] for two noncommuting operators $\hat{a}$ and $\hat{b}$,

$$\exp(\hat{a})\exp(\hat{b}) = \exp\left(\hat{a}+\hat{b}+\frac{1}{2}[\hat{a},\hat{b}]+\frac{1}{12}[\hat{a}-\hat{b},[\hat{a},\hat{b}]]+\cdots\right), \qquad (12)$$

where $[\hat{a},\hat{b}] = \hat{a}\hat{b}-\hat{b}\hat{a}$. A comparison of Equations (9) and (11) shows that the Split-Step Fourier Method ignores the noncommutating nature of the operators $\hat{D}$ and $\hat{N}$. By using Equation (12) with $\hat{a} = h\hat{D}$ and $\hat{b} = h\hat{N}$, the dominant error term is found to result from the single commutator $\frac{1}{2}h^2[\hat{D},\hat{N}]$. Thus, the Split-Step Fourier Method is accurate to second order in the step size h [20].

3.3 Symmetrized Split-Step Fourier Method

The accuracy of the Split-Step Fourier Method can be improved by adopting a different procedure to propagate the optical pulse over one segment from z to $z+h$. In this procedure Equation (9) is replaced by

$$A(z+h,T) \approx \exp\left(\frac{h}{2}\hat{D}\right)\exp\left(\int_z^{z+h}\hat{N}(z')dz'\right)\exp\left(\frac{h}{2}\hat{D}\right)A(z,T). \qquad (13)$$

The main difference is that the effect of nonlinearity is included in the middle of the segment rather than at the segment boundary. Because of the symmetric form of the

exponential operators in Equation (13), this scheme is known as the Symmetrized Split-Step Fourier Method (S-SSFM) [22]. The integral in the middle exponential is useful to include the z dependence of the nonlinear operator $\hat{N}$. If the step size h is small enough, it can be approximated by $\exp(h\hat{N})$, similar to Equation (9). The most important advantage of using the symmetrized form of Equation (13) is that the leading error term results from the double commutator in Equation (12) and is of third order in the step size h.

The accuracy of the Split-Step Fourier Method can be improved by evaluating the interal in Equation (13) more accurately than approximating it by $h\hat{N}(z)$. A simple approach is to employ the trapezoidal rule and approximate the integral by [23]

$$\int_z^{z+h} \hat{N}(z')dz' \approx \frac{h}{2}[\hat{N}(z) + \hat{N}(z+h)]. \qquad (14)$$

However, the implementation of Equation (14) is not simple because $\hat{N}(z+h)$ is unknown at the midsegment located at $z + h / 2$. It is necessary to follow an iterative procedure that is initiated by replacing $\hat{N}(z+h)$ by $\hat{N}(z)$. Equation (13) is then used to estimate $A(z+h,T)$ which in turn is used to calculate the new value of $\hat{N}(z+h)$. Although the iteration procedure is time-consuming, it can still reduce the overall computing time if the step size h can be increased because of the improved accuracy of the numerical algorithm. Two iterations are generally enough in practice.

The implementation of the Symmetrized Split-Step Fourier Method is relatively straightforward. As shown in Figure 4, the fiber length is divided into a large number of

segments that need not be spaced equally. The optical pulse is propagated from segment to segment using the prescription of Equation (13). More specifically, the optical field $A(z,T)$ is first propagated for a distance h / 2 with dispersion only using the FFT algorithm and Equation (10). At the midplane $z + h / 2$, the field is multiplied by a nonlinear term that represents the effect of nonlinearity over the whole segment length h. Finally, the field is propagated the remaining distance $h / 2$ with dispersion only to obtain $A(z+h,T)$. In effect, the nonlinearity is assumed to be lumped at the midplane of each segment(dashed lines in Figure 4).

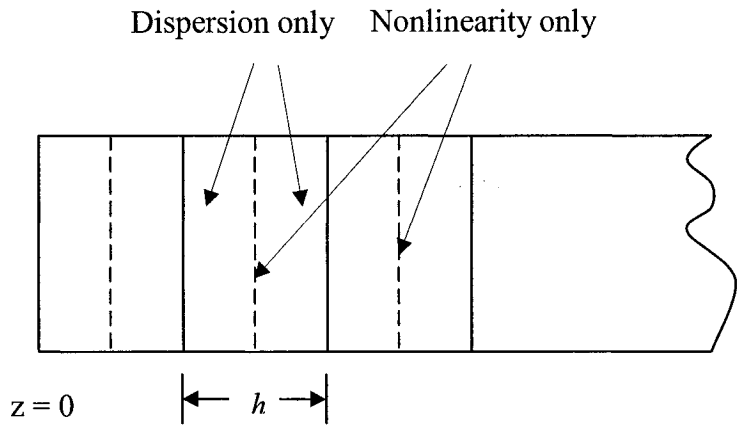

Figure 4. Schematic illustration of the Symmetrized Split-Step Fourier Method used for numerical simulations. Fiber length is divided into large number of segments of width h. With in a segment, the effect of nonlinearity is included at the midplane shown by a dashed line.

To be clear, we list the specific procedures in details.

The Symmetrized Split-Step Fourier Method:

The first step:

$$A(0+h,T)=\exp(\frac{h}{2}\hat{D})\exp[\frac{h}{2}\hat{N}(0)+\frac{h}{2}\hat{N}(0+h)]\exp(\frac{h}{2}\hat{D})A(0,T), \quad (15)$$

where $A(0,T)$ is the initial field.

From Equation (8), $\hat{N}(0)$ of Equation (15) is easy to acquire, but it's $\hat{N}(0+h)$ is not known right now, because $A(0+h,T)$ is needed to calculate $\hat{N}(0+h)$.

So, an approximation of $\hat{N}(0+h)$ is obtained by using the following way.

First of all,

$$\int_z^{z+h}\hat{N}(z')dz'\approx\frac{h}{2}[\hat{N}(z=0)+\hat{N}(z=0)] \quad (16)$$

Secondly, substitute Equation (16) into Equation (13),

then we have

$$\overline{A}(0+h,T)=\exp(\frac{h}{2}\hat{D})\exp[\frac{h}{2}\hat{N}(0)+\frac{h}{2}\hat{N}(0)]\exp(\frac{h}{2}\hat{D})A(0,T). \quad (17)$$

Now, $\overline{A}(0+h,T)$ is ready for estimating $\hat{N}(0+h)$.

From Equation (8),

$$\hat{N}(0+h)=i\gamma\left\{(1-f_R)\left(\left|\overline{A}(0+h,T)\right|^2+\frac{i}{\omega_0}\frac{1}{\overline{A}(0+h,T)}\frac{\partial}{\partial T}(\overline{A}(0+h,T)\left|\overline{A}(0+h,T)\right|^2)\right)\right\}$$

$$+ f_R\left[h_R(T) * \left|\overline{A}(0+h,T)\right|^2 + \frac{i}{\omega_0}\frac{1}{\overline{A}(0+h,T)}\frac{\partial}{\partial T}\left(\overline{A}(0+h,T)h_R(T) * \left|\overline{A}(0+h,T)\right|^2\right)\right] \tag{18}$$

Finally, $A(0+h,T)$ can be acquired.

$$A(0+h,T) = \exp(\frac{h}{2}\hat{D})\exp[\frac{h}{2}\hat{N}(0) + \frac{h}{2}\hat{N}(0+h)]\exp(\frac{h}{2}\hat{D})A(0,T) \tag{19}$$

The second step:

To acquire $A(h+h,T)$, we need to calculate $\hat{N}(h)$ and $\hat{N}(h+h)$.

From the First step, we have already known $\hat{N}(h)$, so only $\hat{N}(h+h)$ should be estimated.

Before estimating $\hat{N}(h+h)$, $\overline{A}(h+h,T)$ is acquired by using Equation (20) as following

$$\overline{A}(h+h,T) = \exp(\frac{h}{2}\hat{D})\exp[\frac{h}{2}\hat{N}(h) + \frac{h}{2}\hat{N}(h)]\exp(\frac{h}{2}\hat{D})A(h,T). \tag{20}$$

In Equation (20), $A(h,T)$ and $\hat{N}(h)$ are from the First step.

By using Equation (20) in Equation (8), $\hat{N}(h+h)$ can be written as

$$\hat{N}(h+h) = i\gamma\left\{(1-f_R)\left(\left|\overline{A}(h+h,T)\right|^2 + \frac{i}{\omega_0}\frac{1}{\overline{A}(h+h,T)}\frac{\partial}{\partial T}(\overline{A}(h+h,T)\left|\overline{A}(h+h,T)\right|^2)\right)\right\}$$

$$+ f_R\left[h_R(T) * \left|\overline{A}(h+h,T)\right|^2 + \frac{i}{\omega_0}\frac{1}{\overline{A}(h+h,T)}\frac{\partial}{\partial T}\left(\overline{A}(h+h,T)h_R(T) * \left|\overline{A}(h+h,T)\right|^2\right)\right] \tag{21}$$

Therefore, $A(h+h,T)$ is finally given by

$$A(h+h,T) = \exp(\frac{h}{2}\hat{D})\exp[\frac{h}{2}\hat{N}(h) + \frac{h}{2}\hat{N}(h+h)]\exp(\frac{h}{2}\hat{D})A(h,T). \quad (22)$$

To obtain the final numerical result, the Second step is repeated the times of amount of segments minus 2.

3.4 Higher Order Symmtrized Split-Step Method

To improve the accuracy of standard Symmetrized Split-Step Method, Higher Order Symmetrized Split-Step Method is chosen in our numerical scheme. In the higher order scheme, the leading error term is cancelled out by a form of extrapolation. The method is simple: we take four forward steps of length h, followed by one backward step of length 2h, then four more forward steps of length h. Then the leading order term in the accumulated error is proportional to $4h^3 + (-2h)^3 + 4h^3$, it cancels out at this order, leaving the leading term in the error per step proportional to h^5. This scheme is therefore globally fourth-order accurate [17].

But, note that we have taken, in effect, nine steps to propagate forward a total distance of $6h$. So, this error extrapolation technique will introduce an overhead.

3.5 Prodictor-Corrector Symmetrized Split-Step Method

Predictor-Corrector Method is a multistep method, because it requires many previous results in the calculation of the next point. By using the combinations of Adams-Bashforth predictor and Adams-Moulton corrector, the accuracy of numerical result of Nonlinear Schrödinger Equation (NLS) can be improved [24]. Fornberg and Driscoll proposed a fast spectral algorithm based on the explicit Adams-Bashforth for the nonlinear part $\hat{N}$ - operator and implicit Adams-Moulton for linear part $\hat{D}$ - operator of Nonlinear Schrödinger Equation [25].

Employing the Predictor-Corrector Method, Equation (5) can be solved as Predictor

$$\overline{A(z_{j+1},T)} = \exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(P_j\hat{N}_j + P_{j-1}\hat{N}_{j-1} + \cdots + P_{j-n+1}\hat{N}_{j-n+1})\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_j,T), \quad (23)$$

where coefficients $P_j, P_{j-1}, \cdots$, and P_{j-n+1} shown in Table 1 [26] are dependent on the order number n of Predictor-Corrector Method. In Equation (23), $\hat{N}_j = \hat{N}(z_j)$.

Table 1. Coefficients P for the different order number n of PC-S-SSFM

Order number n	P_j	P_{j-1}	P_{j-2}	P_{j-3}	P_{j-4}
2	3/2	-1/2			
3	23/12	-16/12	5/12		
4	55/24	-59/24	37/24	-9/24	
5	1901/720	-2774/720	2616/720	-1274/720	251/720

The estimated $\overline{A(z_{j+1},T)}$ in turn is used to calculate the new value of $\overline{\hat{N}}_{j+1}$ by using Equation (8) as

$$\overline{\hat{N}}_{j+1} = i\gamma\left\{(1-f_R)\left(\left|\overline{A(z_{j+1},T)}\right|^2 + \frac{i}{\omega_0}\frac{1}{\overline{A(z_{j+1},T)}}\frac{\partial}{\partial T}\left(\overline{A(z_{j+1},T)}\left|\overline{A(z_{j+1},T)}\right|^2\right)\right)\right\}$$

$$+ f_R\left[h_R(T) * \left|\overline{A(z_{j+1},T)}\right|^2 + \frac{i}{\omega_0}\frac{1}{\overline{A(z_{j+1},T)}}\frac{\partial}{\partial T}\left(\overline{A(z_{j+1},T)}h_R(T) * \left|\overline{A(z_{j+1},T)}\right|^2\right)\right] \quad (24)$$

Finally Equation (5) is solved as

$$A(z_{j+1},T) = \exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(C_{j+1}\overline{\hat{N}}_{j+1} + C_j\overline{\hat{N}}_j + \cdots + C_{j-n+1}\overline{\hat{N}}_{j-n+1})\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_j,T), \quad (25)$$

where coefficients $C_{j+1}, C_j, \cdots$ and C_{j-n+1} shown in Table 2 [26] are dependent on the order number n of Predictor-Corrector Method.

Table 2. Coefficients C for the different order number n of PC-S-SSFM

Order number n	C_{j+1}	C_j	C_{j-1}	C_{j-2}	C_{j-3}	C_{j-4}
2	1/4	2/4	1/4			
3	5/24	13/24	7/24	-1/24		
4	9/48	28/48	14/48	-4/48	1/48	
5	251/1440	897/1440	382/1440	-158/1440	87/1440	-19/1440

In Equations (24) and (25), $\overline{\hat{N}}_j = \overline{\hat{N}}(z_j)$ and $\overline{\hat{N}}_{j+1} = \overline{\hat{N}}(z_j + h)$.

Also, to be clear, we chose the 4th order Predictor-Corrector S-SSFM as an example and list the specific procedures in details.

The first step:

First of all, the fiber is initiated by $A(z_j,T) = A(z_0,T)$ when $j = 0$.

$$\overline{A(z_1,T)} = \exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(\frac{55}{24}\hat{N}_0)\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_0,T) \tag{26}$$

In Equation (26), $\hat{N}_0$ can be obtained by using Equation (8)

$$\hat{N}_0 = i\gamma\left\{(1-f_R)\left(|A(z_0,T)|^2 + \frac{i}{\omega_0}\frac{1}{A(z_0,T)}\frac{\partial}{\partial T}(A(z_0,T)|A(z_0,T)|^2)\right)\right\}$$

$$+ f_R\left[h_R(T) * |A(z_0,T)|^2 + \frac{i}{\omega_0}\frac{1}{A(z_0,T)}\frac{\partial}{\partial T}\left(A(z_0,T)h_R(T) * |A(z_0,T)|^2\right)\right] \tag{27}$$

Second, by substituting Equation (26) into Equation (24), we have

$$\overline{\hat{N}}_1 = i\gamma\left\{(1-f_R)\left(\left|\overline{A(z_1,T)}\right|^2 + \frac{i}{\omega_0}\frac{1}{\overline{A(z_1,T)}}\frac{\partial}{\partial T}(\overline{A(z_1,T)}\left|\overline{A(z_1,T)}\right|^2)\right)\right\}$$

$$+f_R\left[h_R(T)*\left|\overline{A(z_1,T)}\right|^2+\frac{i}{\omega_0}\frac{1}{\overline{A(z_1,T)}}\frac{\partial}{\partial T}\left(\overline{A(z_1,T)}h_R(T)*\left|\overline{A(z_1,T)}\right|^2\right)\right] \tag{28}$$

Finally, the corrector $A(z_1,T)$ is acquired.

$$A(z_1,T)=\exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(\frac{9}{48}\overline{\hat{N}}_1)\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_0,T) \tag{29}$$

The second step:

By using the result of the first step Equation (29), $\hat{N}_1$ can be estimated as

$$\hat{N}_1=i\gamma\left\{(1-f_R)\left(\left|A(z_1,T)\right|^2+\frac{i}{\omega_0}\frac{1}{A(z_1,T)}\frac{\partial}{\partial T}(A(z_1,T)\left|A(z_1,T)\right|^2)\right)\right\}$$

$$+f_R\left[h_R(T)*\left|A(z_1,T)\right|^2+\frac{i}{\omega_0}\frac{1}{A(z_1,T)}\frac{\partial}{\partial T}\left(A(z_1,T)h_R(T)*\left|A(z_1,T)\right|^2\right)\right] \tag{30}$$

So, the next predictor is

$$\overline{A(z_2,T)}=\exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(\frac{55}{24}\hat{N}_1+\frac{-59}{24}\hat{N}_0)\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_1,T) \tag{31}$$

Using the predictor Equation (31), $\overline{\hat{N}}_2$ can be found as

$$\overline{\hat{N}_2} = i\gamma\left\{(1-f_R)\left(\left|\overline{A(z_2,T)}\right|^2 + \frac{i}{\omega_0}\frac{1}{\overline{A(z_2,T)}}\frac{\partial}{\partial T}(\overline{A(z_2,T)}\left|\overline{A(z_2,T)}\right|^2)\right)\right\}$$

$$+ f_R\left[h_R(T) * \left|\overline{A(z_2,T)}\right|^2 + \frac{i}{\omega_0}\frac{1}{\overline{A(z_2,T)}}\frac{\partial}{\partial T}\left(\overline{A(z_2,T)}h_R(T) * \left|\overline{A(z_2,T)}\right|^2\right)\right] \quad (32)$$

So, the corrector is obtained as

$$A(z_2,T) = \exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(\frac{9}{48}\overline{\hat{N}}_2 + \frac{28}{48}\overline{\hat{N}}_1)\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_1,T) \quad (33)$$

To be simple, we just give the corresponding Predictors and Correctors of the next three steps.

The third step:

The predictor:

$$\overline{A(z_3,T)} = \exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(\frac{55}{24}\hat{N}_2 + \frac{-59}{24}\hat{N}_1 + \frac{37}{24}\hat{N}_0)\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_2,T) \quad (34)$$

The corrector:

$$A(z_3,T) = \exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(\frac{9}{48}\overline{\hat{N}}_3 + \frac{28}{48}\overline{\hat{N}}_2 + \frac{14}{48}\overline{\hat{N}}_1)\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_2,T) \quad (35)$$

The fourth step:

The predictor:

$$\overline{A(z_4,T)} = \exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(\frac{55}{24}\hat{N}_3 + \frac{-59}{24}\hat{N}_2 + \frac{37}{24}\hat{N}_1 + \frac{-9}{24}\hat{N}_0)\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_3,T) \quad (36)$$

The corrector:

$$A(z_4,T) = \exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(\frac{9}{48}\overline{\hat{N}}_4 + \frac{28}{48}\overline{\hat{N}}_3 + \frac{14}{48}\overline{\hat{N}}_2 + \frac{-4}{48}\overline{\hat{N}}_1)\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_3,T) \quad (37)$$

The fifth step:

The predictor:

$$\overline{A(z_5,T)} = \exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(\frac{55}{24}\hat{N}_5 + \frac{-59}{24}\hat{N}_4 + \frac{37}{24}\hat{N}_3 + \frac{-9}{24}\hat{N}_2)\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_4,T) \quad (38)$$

The corrector:

$$A(z_5,T) = \exp\left(\frac{h\hat{D}}{2}\right)\exp\left[h(\frac{9}{48}\overline{\hat{N}}_5 + \frac{28}{48}\overline{\hat{N}}_4 + \frac{14}{48}\overline{\hat{N}}_3 + \frac{-4}{48}\overline{\hat{N}}_2 + \frac{1}{48}\overline{\hat{N}}_1)\right]\exp\left(\frac{h\hat{D}}{2}\right)A(z_4,T)$$

(39)

The above five main steps clearly show the Predictor-Corrector-Symmetrized-Split-Step Fourier Method, and just following the same rules, the final result can be easily found.

Chapter 4

Simulations and Comparisons

4.1 The Platform of Simulations

We choose Matlab Development Environment shown in Figure 5 as our platform of simulations because MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows us to solve many technical computing problems, especially those with matrix and vector formulations. Also, Matlab is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation.

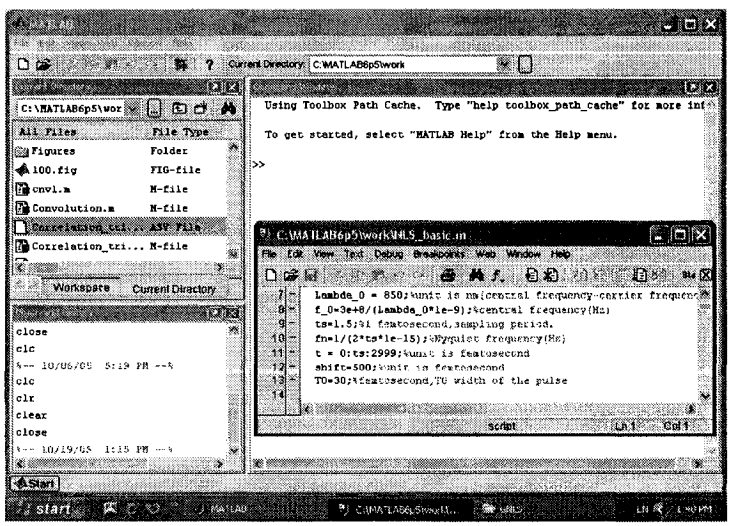

Figure 5. The desktop of Matlab Development Environment.

4.2 The Background of Simulations

Photonic Crystal Fibers show great potential for various applications in the field of optical frequency metrology, sensor technology, and optical telecommunications [7], [9], [27], [28]. These fibers consist of a solid silica core surrounded by an array of air holes running along the fiber. The special structure provides a wavelength-dependent effective index for the cladding and can allow single-mode guidance throughout the visible and near infrared [29]. By varying the arrangement and size of the holes, the dispersion properties of the fiber can be tailored in broad ranges and, for instance, the effective area of the propagating mode can be adjusted to enhance the optical nonlinearities in the fiber. The combination of the unique dispersion properties and enhanced nonlinearities can be used for advantage to obtain efficient generation of supercontinuum when the fiber is pumped with short pulses of laser light with wavelength located in the vicinity of the zero-dispersion wavelength.

To simulate the generation of supercontinuum through photonic crystal fibers, a schematic setup is depicted in Figure 6. The output pulse train of a mode-locked Ti:Sapphire laser (Tsunami/Spectra Physics) is coupled into a PCF.

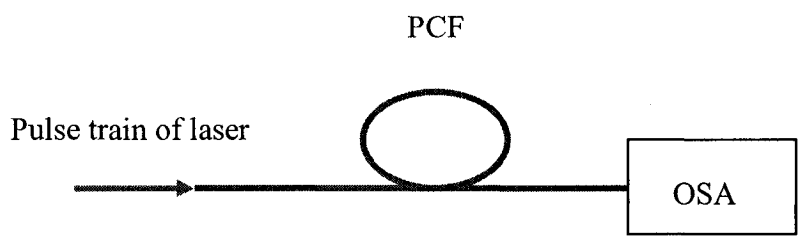

Figure 6. The schematic setup of generation of supercontinuum through PCF.

The laser produces 100 femtoseconds pulses (FWHM) at a repetition rate of 80 MHz and with the peak power of 10 kW. The optical spectrum is recorded at the fiber output by using an optical spectrum analyzer (OSA).

Because the purpose of our simulations is to find a faster method in the simulations of generation of supercontinuum, what we need is only to use some parameters of the photonic crystal fibers that are already found to have the property to generate the supercontinuum. The corresponding parameters borrowed from reference [30] are as following:

$\beta_2 = -1.276\times10^{-2}\, ps^2\cdot m^{-1}$, $\beta_3 = 8.119\times10^{-5}\, ps^3\cdot m^{-1}$, $\beta_4 = -1.321\times10^{-7}\, ps^4\cdot m^{-1}$,

$\beta_5 = 3.032\times10^{-10}\, ps^5\cdot m^{-1}$, $\beta_6 = -4.196\times10^{-13}\, ps^6\cdot m^{-1}$, $\beta_7 = 2.570\times10^{-16}\, ps^6\cdot m^{-1}$,

$\gamma = 0.0452\, 1/W\cdot m$, $\omega_0 = 850nm$.

We choose $\alpha = 0$, because the length of the PCF is only 10 cm.

Initial field is expressed as

$A(0,T) = \sqrt{P_0}\sec\left(\frac{T}{T_0}\right)$, $T_{FWHM} \approx 1.763\, T_0$, $P_0 = 10$ kW, P_0 is the peak power,

$T_{FWHM} = 100$ fs.

4.3 The Simulation by Using the Higher Order Symmetrized Split-Step Method

In the simulation of Higher Order Symmetrized Split-Step Method, the optical field expressed by using peak power 10 kW hyperbolic-secant pulse with $T_{FWHM} = 100$ fs is pumped into the PCF of 10 cm length. The practical parameters of the PCF are given in

Chapter 4. Section 4.2. To make sure the generation of supercontinuum can be obtained, 2^{11} points are necessary in the windows of time and spectrum. Forward step of length h is 10 μm, and backward step length is 20 μm. In higher order scheme, nine steps propagate forward a total distance of six steps, so total 15000 steps are needed to propagate 10 cm distance of the photonic crystal fiber. The Matlab program of the Higher Order Symmetrized Split-Step Fourier Method is given in Appendix I. In Figure 7, the optical fields in time domain and frequency domain before and after the propagation through the photonic crystal fiber are plotted. As shown in Figure 7, (a) the one on the left top is the optical pulse before the propagation, and (c) the right top one is the spectrum of the initial optical pulse; (b) on the bottom of left, the picture shows the shaped pulse at the end of the PCF, and (d) the one on the right bottom is the broadened spectrum.

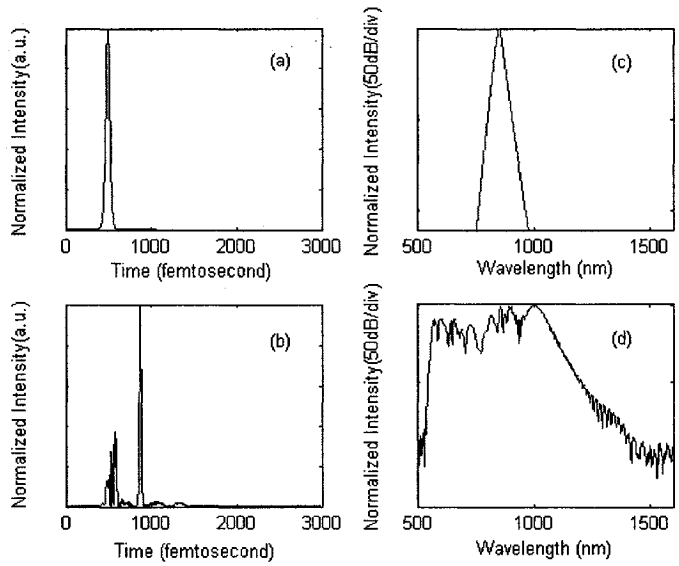

Figure 7. The optical fields in time domain and frequency domain before and after the propagation through the PCF in the simulation of HO-S-SSFM. (a) The input pulse in time domain; (b) The output pulse in time domain; (c) The spectrum of the input pulse; (d) The spectrum of the output pulse.

To be clear, we show the spectrums of the pulse before and after broadening in Figure 8 and Figure 9. It is obvious to see that the spectrum of the pulse is drastically broadened.

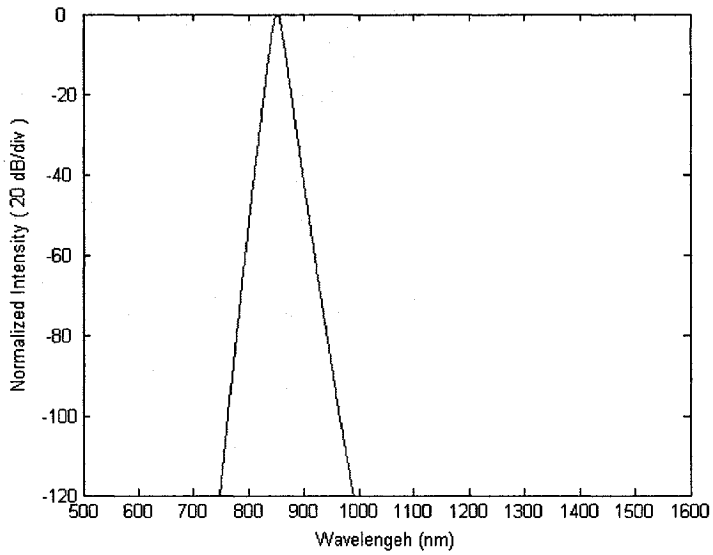

Figure 8. The spectrum of the initial pulse in the simulation of HO-S-SSFM.

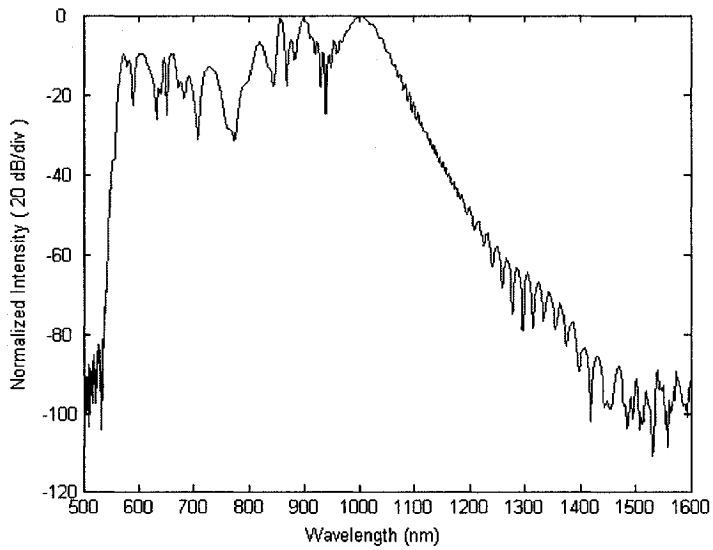

Figure 9. The spectrum of the pulse after propagation through 10cm PCF in the simulation of HO-S-SSFM.

4.4 The Simulation by Using the Predictor-Corrector Symmetrized Split-Step Method

In the simulation of Predictor-Corrector Symmetrized Split-Step Method, the fifth order PC-S-SSFM is employed. The optical field expressed by using peak power 10 kW hyperbolic-secant pulse with T_{FWHM} = 100 fs is pumped into the PCF of 10 cm length. The practical parameters of the PCF are the same with those given in Chapter 4. Section 4.2. To make sure the generation of supercontinuum can be obtained, 2^{11} points are also necessary in the windows of time and spectrum. Forward step of length h is 10 μm. In the Predictor-Corrector Symmetrized Split-Step Fourier Method, one step propagates forward a distance of one step, so total 10000 steps are needed to propagate 10 cm distance of the PCF. The Matlab program of the Predictor-Corrector Symmetrized Split-Step Fourier Method can be found in Appendix II. In Figure 10, the optical fields in the time domain and the frequency domain before and after the propagation through the PCF are plotted. As shown in Figure 10, the one on the left top (a) is the optical pulse without any propagation, and the right top one (c) is the spectrum of the initial optical field; on the bottom of the left (b), the picture shows the shaped pulse at the end of the PCF of 10cm length, and the one on the right bottom (d) is the broadened spectrum.

Also, we especially give the Figure 11 and Figure 12 to show how the two spectrums look like and how much the spectrum of the initial pulse becomes wider after it goes out through the PCF of 10cm length.

By comparing Figure 11 and Figure 12, the concept of the spectral supercontinuum generation will be easy to find in our simulations.

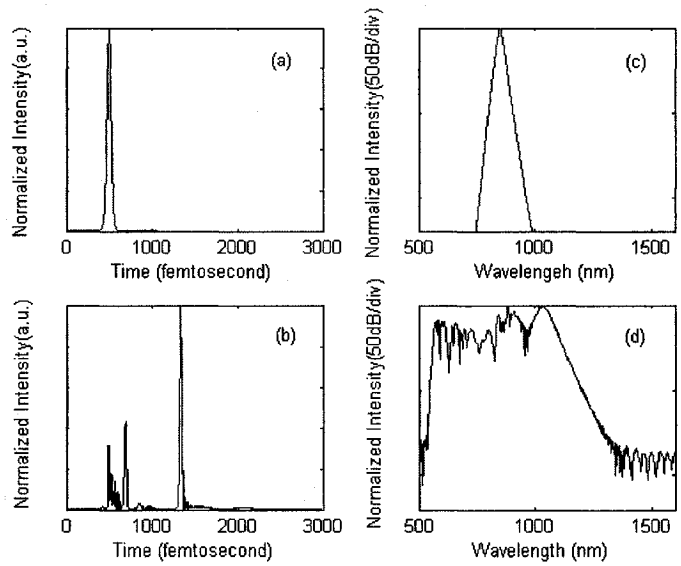

Figure 10. The optical fields in time domain and frequency domain before and after the propagation through the PCF in the simulation of PC-S-SSFM. (a) The input pulse in time domain; (b) The output pulse in time domain; (c) The spectrum of the input pulse; (d) The spectrum of the output pulse.

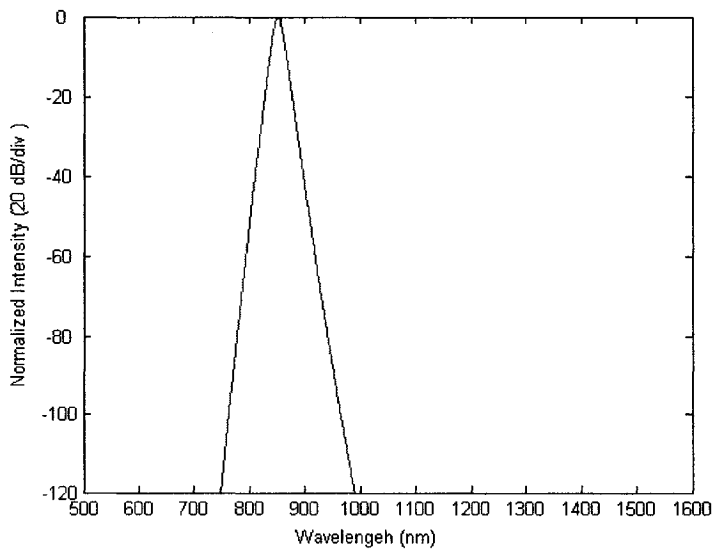

Figure 11. The spectrum of initial pulse in the simulation of PC-S-SSFM.

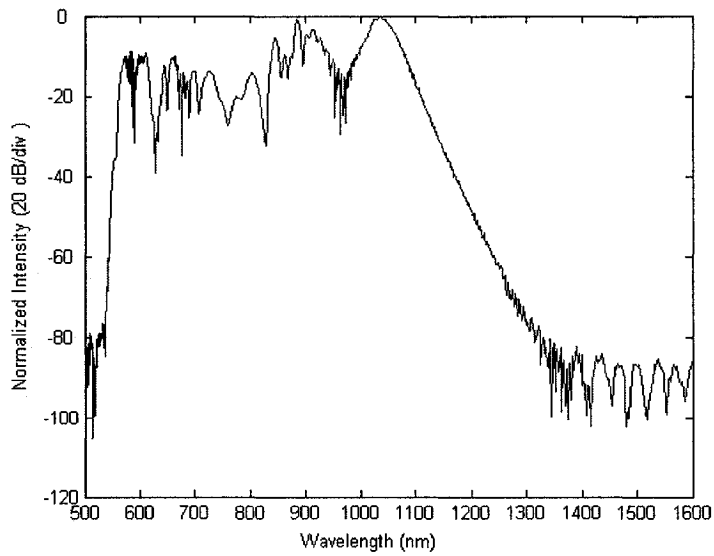

Figure 12. The spectrum of the pulse after propagation through 10cm PCF in the simulation of PC-S-SSFM.

4.5 The Simulations of the Evolution of Supercontinuum Generation

In the area of fiber source, to understand the underlying mechanisms leading to the supercontinuum generation, the evolution of SC generation plays a very important role. So, in our simulations we give the detailed information of how the optical field develops in the time domain and the frequency domain.

Peak power 10 kW hyperbolic-secant pulse with T_{FWHM} = 100 fs is pumped into the PCF of 10 cm length. Step size is 10 μm. The practical parameters of the PCF are the same with those given in Chapter 4. Section 4.2.

4.5.1 The Simulations of the Evolution of SC by Using HO-S-SSFM

Actually a dynamic process of development of SC can be acquired in our simulations, but in this thesis our intention is focused on efficiency of simulation method. So we do not have to show the whole process of the evolution of SC, and some observation points are chosen to demonstrate the evolution of SC. Typical observing points are selected at the points of 0cm, 1cm, 2cm, 3cm, 4cm, 5cm, 6cm, 7cm, 8cm, 9cm and 10cm length. The corresponding program is given in Appendix III. In Figure 13 and Figure 14, the simulations of evolution are given in the frequency domain and in the time domain. In each figure, the curve is organized in the sequence of advanced direction, and the advanced direction is from the bottom to the top.

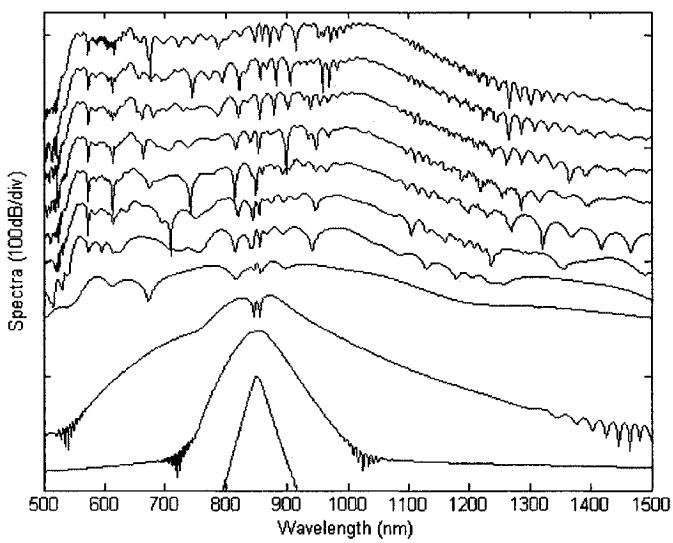

Figure 13. The evolution of spectrum is shown at the points of 0cm to 10cm with 1μm step size by using HO-S-SSFM.

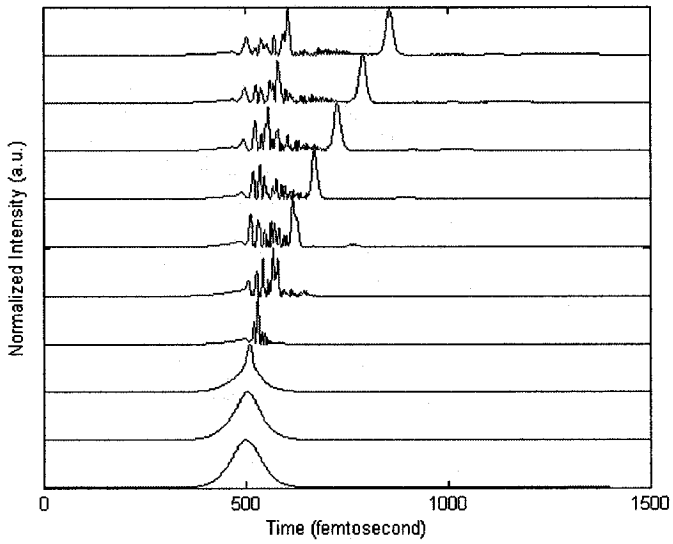

Figure 14. The evolution of the pulse in the time domain is shown at the points of 0cm to 10cm with 1 μm step size by using HO-S-SSFM.

4.5.2 The Simulations of the Evolution of SC by Using PC-S-SSFM

Typical observing points are selected at the points of 0cm, 1cm, 2cm, 3cm, 4cm, 5cm, 6cm, 7cm, 8cm, 9cm and 10cm length. The corresponding program is given in Appendix IV. In Figure 15 and Figure 16, the simulations of evolution are given in the frequency domain and in the time domain. In each figure, the curve is organized in the sequence of advanced direction, and the advanced direction is from the bottom to the top.

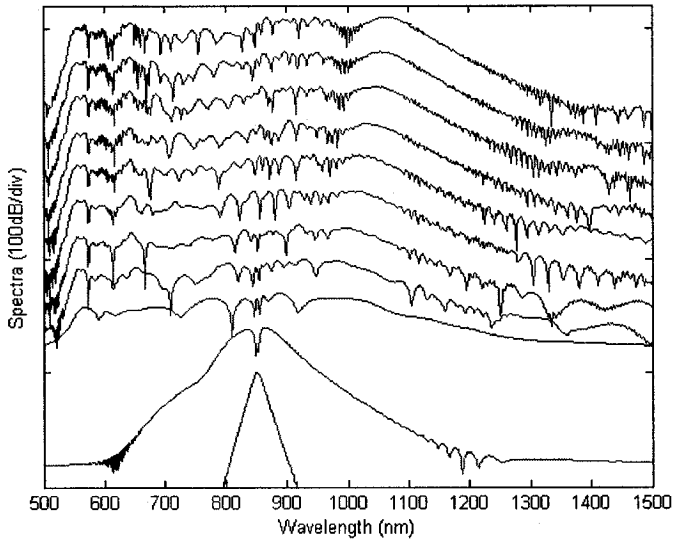

Figure 15. The evolution of spectrum is shown at the points of 0cm to 10cm with 1μm step size by using PC-S-SSFM.

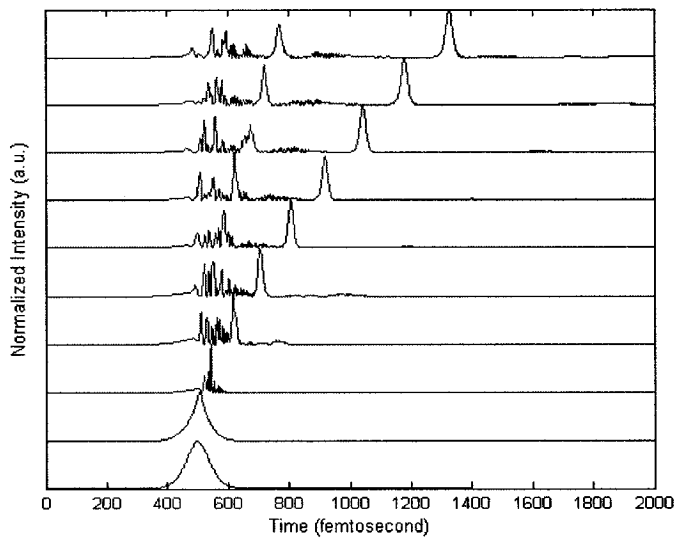

Figure 16. The evolution of the pulse in the time domain is shown at the points of 0cm to 10cm with 1μm step size by using PC-S-SSFM.

4.6 The Comparisons of the Simulations

In the past few years, supercontinuum generation in photonic crystal fibers has become a subject of intense worldwide study. But in most publications, supercontinuum generations are focused on experimental study. Although the simulations of SC generation are mentioned in some of publications, they never mentioned the very basic parameters and even which methods they used. We read a lot of references, and finally we found an important publication that could help us to verify the accuracy of our simulations.

The reference [30] is a very typical publication in simulations of supercontinuum generation. Very detailed parameters and method used in simulations of SC can be found in that paper. So we borrowed the physical parameters and the result of their simulations to testify what we did.

4.6.1 The Results of Simulations in the Reference [30]

In the reference [30], the effect of dispersion is described by using terms up to β_7.

Particular values at $\omega_0 = 850nm$ are $\beta_2 = -1.276\times10^{-2}\, ps^2\cdot m^{-1}$,

$\beta_3 = 8.119\times10^{-5}\, ps^3\cdot m^{-1}$, $\beta_4 = -1.321\times10^{-7}\, ps^4\cdot m^{-1}$, $\beta_5 = 3.032\times10^{-10}\, ps^5\cdot m^{-1}$,

$\beta_6 = -4.196\times10^{-13}\, ps^6\cdot m^{-1}$, $\beta_7 = 2.570\times10^{-16}\, ps^6\cdot m^{-1}$, $\gamma = 0.0452\, 1/W\cdot m$.

They choose $\alpha = 0$, because the length of the PCF is only 10 cm.

Initial field is expressed as

$A(0,T) = \sqrt{P_0}\,\mathrm{sec}\left(\frac{T}{T_0}\right)$, $T_{FWHM} \approx 1.763\, T_0$, $P_0 = 10$ kW, P_0 is the peak power,

T_{FWHM} = 50 fs.

The numerical method to solve the propagation Equation (1) is the so called Enhanced Split-Step Fourier Method [30], which is the similar with the Higher Order Symmetrized Split-Step Method we use. Spectral evolution at the points of 0cm, 0.5cm, 1cm, 2cm, 4cm, 6cm, 8cm and 10cm along the PCF is shown in Figure 17 [30].

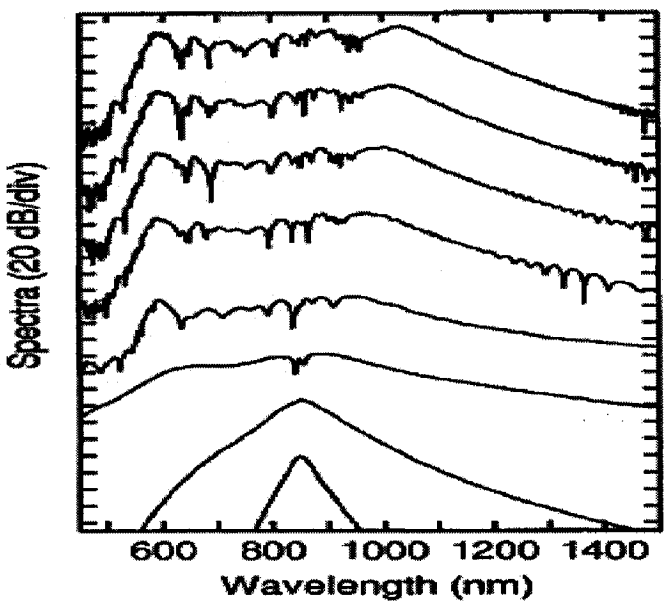

Figure 17. Evolution of 50 fs input pulse along 10cm of PCF[30].

4.6.2 The Results of Simulations of the HO-S-SSFM

To compare the results of simulations of HO-S-SSFM with the published results, we choose the exact parameters used in the reference [30] and the simulation results are shown in Figure 18.

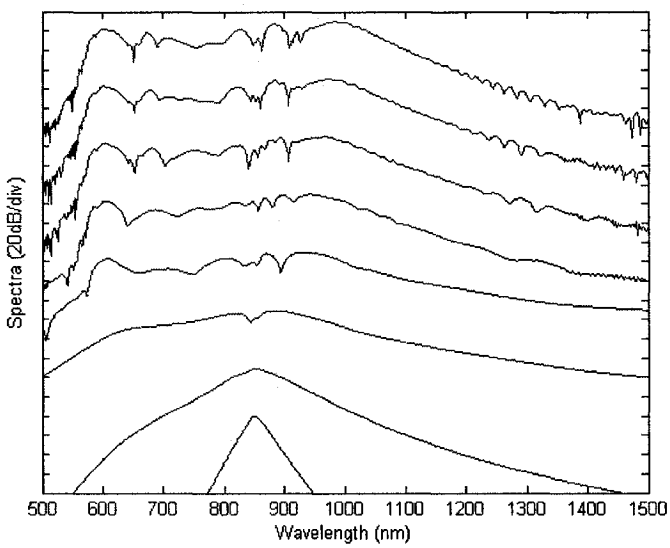

Figure 18. Evolution of 50 fs input pulse along 10cm of PCF by using HO-S-SSFM.

As we have seen in Figures 17 and 18, very good agreements between the two results of simulations can be found.

First of all, drastic spectral broadening is observed in both Figures 17 and 18. Each output spectra of simulations extends over an octave 550-1100 nm.

Second, both input pulses experience significantly more spectral broadening in the first few centimeters of propagation. For example, drastic broadenings both occur after 2cm propagation.

Third, it is easy to find that both initial stages of propagation are associated with approximately symmetrical spectral broadenings centered at $\omega_0 = 850nm$ and the symmetries are broken after 2cm propagation.

Finally, the spectral intensities both exhibit a somewhat complex structure with particularly distinct peaks on along the spectrums although the output spectra are essentially continuous over this range.

By the comparison, we think that our simulations of the HO-S-SSFM are correct and accurate.

4.6.3 The Results of Simulations of the PC-S-SSFM

Also, to compare the results of simulations of PC-S-SSFM with the published results, we choose the exact parameters used in the reference [30] and the simulation results are shown in Figure 19.

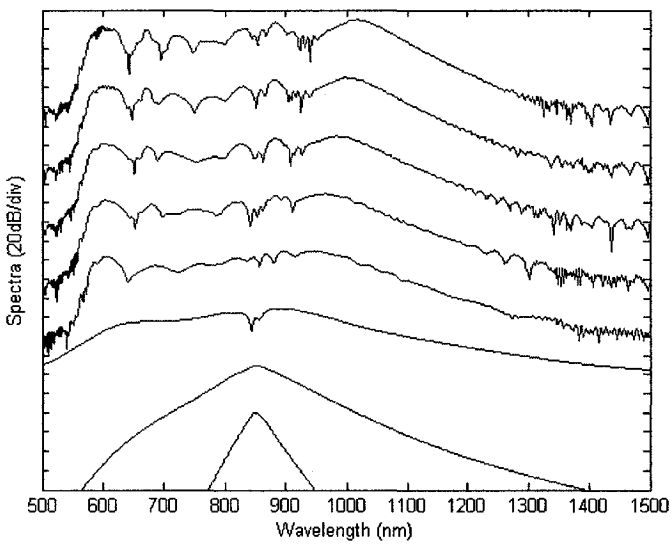

Figure 19. Evolution of 50 fs input pulse along 10cm of PCF by using PC-S-SSFM.

As we have seen in Figures 17 and 19, very good agreements between the two results of simulations can be found.

First of all, drastic spectral broadening is observed in both Figures 17 and 19. Each output spectra of simulations extends over an octave 550-1100 nm.

Second, both input pulses experience significantly more spectral broadening in the first few centimeters of propagation. For example, drastic broadenings both occur after 2cm propagation.

Third, it is easy to find that both initial stages of propagation are associated with approximately symmetrical spectral broadenings centered at $\omega_0 = 850nm$ and the symmetries are broken after 2cm propagation.

Finally, the spectral intensities both exhibit a somewhat complex structure with particularly distinct peaks on along the spectrums although the output spectra are essentially continuous over this range.

By the comparison, we show that the simulations of PC-S-SSFM are correct and accurate.

4.7 The Comparisons of the HO-S-SSFM and the PC-S-SSFM

In Sections 4.6.2 and 4.6.3, by the comparisons with the published result, we qualitatively prove that HO-S-SSFM and PC-S-SSFM are both very effective methods in simulations of supercontinuum generation of PCF.

But we must point out that the time used in the two simulation methods is very different although the two simulation results show the same level of accuracy. In this section, we are going to compare the time which the two methods use in the same condition.

4.7.1 The Accuracy Comparison of the HO-S-SSFM and the PC-S-SSFM

Relative errors are used in the accuracy comparison of the HO-S-SSFM and the PC-S-SSFM. The relative error δ is defined by

$$\delta = \frac{\|A\| - \|A_0\|}{\|A_0\|} \tag{40}$$

where the norm $\|A\|$ is defined as $\|A\| = \sqrt{\int |A(T)|^2 dT}$, and A and A_0 are the numerical and the exact values at the end of the PCF, respectively. Because Equation (5) does not have the exact solution in our simulations, A_0 should be numerically acquired when the step size h is enough small. In our comparison, A_0 is obtained when the step size h is 1 μm, and $\|A_0\|$ is the average of $\|A_0\|$ from the HO-S-SSFM and $\|A_0\|$ from the PC-S-SSFM. The PCF's parameters are the same with the previous sections. In Figure 20, the relative errors with different step sizes by using the HO-S-SSFM and the PC-S-SSFM are given. As we have seen in Figure 20, the relative errors of the PC-S-SSFM become smaller than those of the HO-S-SSFM when the step size is 20 μm. And also with the reducing of step size h, the relative errors of the PC-S-SSFM decrease rapidly until the step size reaches 12.5 μm and keep almost the same value when step size is 10 μm. On the other hand, the relative errors of the HO-S-SSFM change very little, but after the step size is less 22.5 μm, they are smaller than the ones of the PC-S-SSFM.

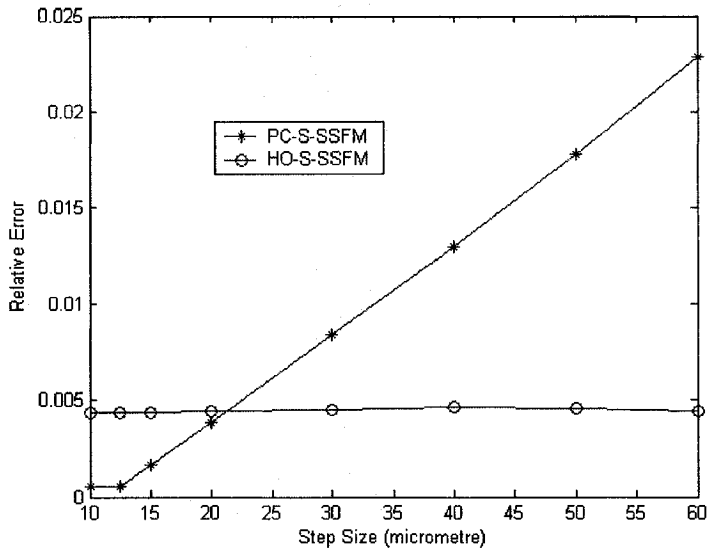

Figure 20. The relative errors with different step sizes by using the HO-S-SSFM and the PC-S-SSFM.

4.7.2 Time Used to Reach Different Points Along the PCF

In the simulations of supercontinuum generation, the evolution of a spectrum is very useful to understand the underlying mechanisms leading to the SC generation. Normally it is unnecessary to acquire the whole process of an evolution, therefore only some typical points are needed to demonstrate the evolution.

As we did before, we use hyperbolic scant input pulses with a peak power of 10 kW and a wavelength of 850 nm. The input pulse parameters are typical of those used in Chapter 4. Section 4.6. The PCF is 10 cm length, and observing points at 0cm, 1cm, 2cm, 3cm, 4cm, 5cm, 6cm, 7cm, 8cm, 9cm and 10cm length of the PCF are chosen. The step length is 10 μm. In Figure 21, the time used by HO-S-SSFM and PC-S-SSFM at different points is plotted.

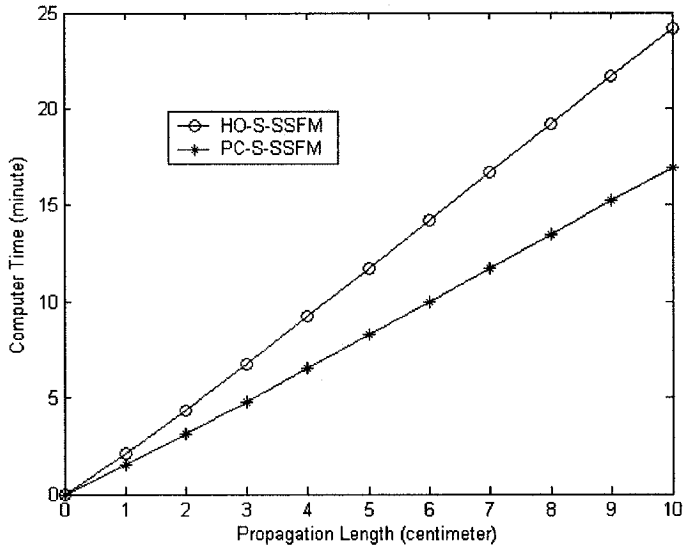

Figure 21. Evolution time used in simulations of HO-S-SSFM & PC-S-SSFM at different points.

Figure 21 illustrates the times used in simulations of HO-S-SSFM and PC-S-SSFM. In the two simulations, evolutions of SC generation are based on the same condition. We separately use HO-S-SSFM and PC-S-SSFM to simulate the evolution of SC generation.

Time is recorded at different points of 0cm, 1cm, 2cm, 3cm, 4cm, 5cm, 6cm, 7cm, 8cm, 9cm and 10cm length of the PCF. As we see in the Figure 21, the time used by the two methods at each same point is different, and at each point the time used by HO-S-SSFM is more than the time used by PC-S-SSFM. And a very clear trend can be seen that the further the pulse propagates along the PCF the more the time is needed by the two methods, but the time of HO-S-SSFM increases faster than the time of PC-S-SSFM. At

the end of the PCF, the HO-S-SSFM costs one thirds more the time needed by the PC-S-SSFM.

4.7.3 Time Used to Reach the End of the PCF with Different Step Sizes

In the simulations of SC generation, the step size is also very important. The amount of step size directly decides how long a simulation will run. So in this section, we are going to compare the PC-S-SSFM with the HO-S-SSFM to see which one is more time-efficient. The two simulations are based on the same condition. A hyperbolic scant pulse with a peak power of 10 kW and the center wavelength of 850 nm is used as the input optical field. The full width of half maximum of the input pulse is 50 fs. The PCF of 10cm length is chosen as the typical one used in our simulations before. By employing the two simulation methods, we select the step size as 10 μm, 12.5 μm, 15μm, 15.5 μm, 20 μm, 25 μm, 30 μm, 40 μm, 50μm and 60 μm. In each simulation, the running time is recorded with different step size when the input pulse reaches the end of the PCF. Figure 22 shows all the time used with different step size.

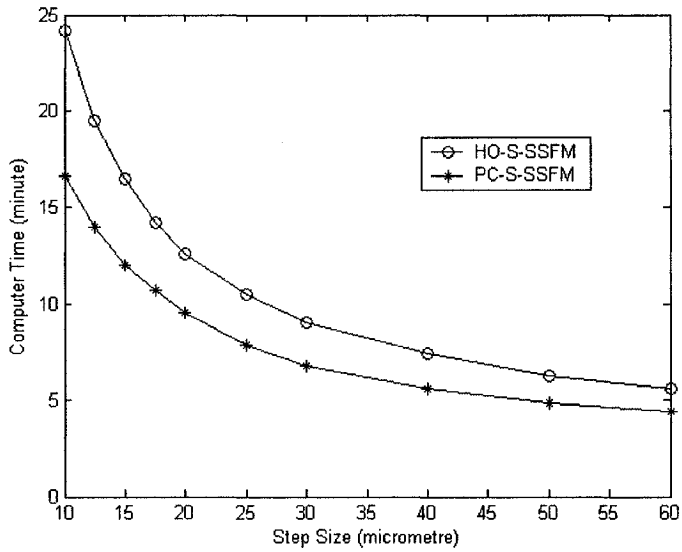

Figure 22. Time used in simulations of HO-S-SSFM & PC-S-SSFM with different steps.

Figure 22 exhibits that the time used in the simulations of the HO-S-SSFM and the PC-S-SSFM with different step size. The horizontal axis gives the step size from 10 μm to 60 μm, and the vertical axis illustrates the time with maximum 25 minutes. When the step size is bigger than 50 μm, the two curves are a little flat. That means times of simulations change very little, but the HO-S-SSFM still costs more time than the PC-S-SSFM. With the step size getting small, for example less 30 μm, both simulation time drastically increase. But the HO-S-SSFM shows more increasing rate than the PC-S-SSFM. With 10 μm step size, the PC-S-SSFM only needs a little bit more than 15 minutes, but the HO-S-SSFM almost reaches 25 minutes. And the trend of the two curves tells that the difference between the time of HO-S-SSFM and PC-S-SSFM will goes far away as the step size becomes smaller.

Chapter 5

Conclusion

In this thesis, we simulated the supercontinuum generation of photonic crystal fiber by using the Higher-Order Symmetrized Split-Step Fourier Method and the Predictor-Corrector Symmetrized Split-Step Fourier Method respectively. Typical physical parameters of a photonic crystal fiber are adopted in our simulations. Based on the same condition, simulation results of the Higher-Order Symmetrized Split-Step Fourier Method, the Predictor-Corrector Symmetrized Split-Step Fourier Method and the published paper are compared, and our simulations are proved to be correct and accurate. To our knowledge, the Predictor-Corrector Symmetrized Split-Step Fourier Method is the first time used in the simulations of supercontinuum generation, and the results of this simulation method are very satisfied. Meanwhile, we compare the time used by the Higher Order Symmetrized Split-Step Fourier Method and the Predictor-Corrector Symmetrized Split-Step Method in two different ways, and the result shows that the Predictor-Corrector Symmetrized Split-Step Method can save almost one third of time used by the Higher Order Symmetrized Split-Step Fourier Method. So we conclude that the Predictor Corrector Symmetrized Split-Step Fourier Method is a new technique to accelerate Symmetrized Split-Step Fourier Method.

REFERENCES

[1] R. R. Alfano and S. L. Shapiro, "Emission in the region 4000 to 7000 Å via four-photon coupling in glass," *Phys. Rev. Lett.*, Vol. 24, No. 11, pp. 584-587, 1970.

[2] P. L. Baldeck and R. R. Alfano, "Intensity effects on the stimulated four photon spectra generated by picosecond pulses in optical fibers," *J. Lightwave Technol.*, Vol. 5, pp. 1712-1715, 1987.

[3] B. P. Nelson, D. Cotter, K. J. Blow, and N. J. Doran, "Large nonlinear pulse broadening in long lengths of monomode fiber," *Opt. Commun.*, Vol. 48, No. 4, pp. 292-294, 1983.

[4] J. K. Ranka, R. S. Windeler, and A. J. Stentz, "Visible continuum generation in air-silica microstructure optical fibers with anomalous dispersion at 800 nm," *Opt. Lett.*, Vol. 25, No. 1, pp. 25-27, 2000.

[5] T. Morioka et al., "1 Tbit/s (100 Gbit/s X 10 channel) OTDM/WDM transmission using a single supercontinuum WDM source," *Electron. Lett.* Vol. 32, pp. 906–907, 1996.

[6] S. T. Cundiff, J. Ye, and J. L. Hall, "Optical frequency synthesis based on mode-locked lasers," *Rev. Sci. Instrum.*, Vol. 72, No. 10, pp. 3749-3771, 2001.

[7] H. Takara et al., "More than 1000 channel optical frequency chain generation from single supercontinuum source with 12.5 GHz channel spacing," *Electron Lett.*, Vol. 36, No. 25, pp. 2089-2090, 2000.

[8] K. Mori, T. Morioka, and M. Saruwatari, "Ultrawide spectral range groupvelocity

dispersion measurement utilizing supercontinuum in an optical fiber pumped by a 1.5 μm compact laser source," *IEEE Trans. Instrum. Meas.*, Vol. 44, pp. 712-715, 1995.

[9] S. A. Diddams et al., "Direct link between microwave and optical frequencies with a 300 THz femtosecond laser comb," *Phys. Rev. Lett.*, Vol. 84, No. 22, pp. 5102-5105, 2000.

[10] S. Coen et al., "Supercontinuum generation by stimulated Raman scattering and parametric four-wave mixing in photonic crystal fibers," *J. Opt. Soc. Am.* B Vol. 19, No 4, 753-764, 2002.

[11] J. M. Dudley et al., "Supercontinuum generation in air–silica microstructured fibers with nanosecond and femtosecond pulse pumping," *J. Opt. Soc. Am.* B Vol. 19, No 4, 765-771, 2002.

[12] J. Herrmann et al., "Experimental Evidence for Supercontinuum Generation by Fission of Higher-Order Solitons in Photonic Fibers," *Phys. Rev. Lett.*, Vol. 88, No. 17, 173901, 2002

[13] M. Lehtonen, " Supercontinuum Generation in Photonic Crystal Fiber," Master's Thesis in Electrical Engineering, in Espoo on the 31st May, 2002

[14] P. Kaiser, E. A. J. Marcatili, and S. E. Miller, "A new optical fiber," *Bell Syst. Tech. J.*, Vol. 52, No. 2, pp. 265-269.

[15] J. Broeng, D. Mogilevstev, S. E. Barkou and A. Bjarklev, "Photonic crystal fibers: a new class of optical waveguides," *Opt. Fiber Technol.*, Vol. 5, pp. 305- 330, 1990.

[16] S. Coen, A. H. L. Chau, R. Leonhardt, J. D. Harvey, J. C. Knight, W. J. Washington, and P. St. J. Russell, "White light supercontinuum generation with 60-ps pump pulse in a photonic crystal fiber," *Opt. Lett.*, vol. 26, pp. 1356-1358, 2001.

[17] K. J. Blow and D. Wood, "Theoretical description of transient stimulated Raman scattering in optical fibers," *IEEE J. Quantum Electron.*, vol.25, pp. 2665-2673, Dec. 1989.

[18] R. H Stolen, J. P. Gordon, W. J. Tomlinson, and H. A. Haus, *J. Opt. Soc. Am.* B 6. 1159, 1989.

[19] P. V. Mamyshev and S. V. Chernikov, *Opt, Lett.*, Vol. 15, pp.1076, 1990.

[20] G. P. Agrawal, *Nonlinear Fiber Optics.* (Academic Press, 2001), Chapter 2.

[21] G. H. Weiss and A. A. Maradudin, *J. Math, Phys.*Vol.3, pp.771, 1962.

[22] J. A. Fleck, J. R. Morris, and M. D. Feit, *Appl. Phys.* Vol. 10, pp. 129, 1976.

[23] M. Lax, J. H. Batteh, and G. P. Agrawal, *J. Appl. Phys.* Vol 52, pp. 109, 1981.

[24] A. Quarteroni, R. Sacco, F. Saleri, *Numerical Mathematics*. New York: Springer-Verlag, 2000.

[25] B. Fornberg, T. A. Driscoll, "A fast spectral algorithm for nonlinear wave equations with linear dispersion," *J. Comp. Phys.*, Vol. 155, pp. 456-467, 1999.

[26] X. M. Liu, B. Lee, "A fast method for nonlinear Schrodinger equation," IEEE *Photon. Technol. Lett.*, Vol. 15, pp. 1549-1551, Nov. 2003.

[27] T. Udem, R. Holzwarth, T. W. Hansch, "Optical frequency metrology," *Nature.* Vol. 416, pp. 233-237, 2002.

[28] T. M. Monro, W. Belardi, K. Furuswa, J. C. Baggett, N. G. Broderick, D. J. Richardson, "Sensing with microstructured optical fibers," *Meas. Sci. Technol.* Vol. 12, pp. 854-858, 2001.

[29] T. A. Briks, J. C. Knight, P. St. J. Russell, "Endlessly single-mode photonic crystal fiber," *Opt. Lett.* Vol. 13, pp. 961-963, 1997.

[30] J. M. Dudley, S. Coen, "Numerical simulations and coherence properties of supercontiuum generation in photonic crystal and tapered optical fibers," *IEEE Qua. Elec.*, Vol. 8, pp 651-659, 2002.

Appendix I

```
clc;%clear command window
clear;
close;
Steps =15000;
P_0 = 10000;%unit W = J/S= J/fs*1e+15
ti = clock;
Lambda_0 = 850;%unit is nm(central frequency-carrier frequency)
f_0=3e+8/(Lambda_0*1e-9);%central frequency(Hz)
ts=1;%1 femtosecond,sampling period.
fn=1/(2*ts*1e-15);%Nyquist frequency(Hz)
t = 0:ts:2999;%unit is femtosecond
shift=500;%unit is femtosecond
T0=56.7;%femtosecond,T0 width of the pulse
A=sqrt(P_0/1e+15)*sech((t-shift)/T0);%Input pulse A(0,t)-(Hyperbolic pulse)
subplot(2,2,1);
plot(t,((abs(A)).^2)/(P_0/1e+15));%amplitudes of the pulse in time domain
title('Input pulse in time domain');
xlabel('Time (femtosecond)');
grid on;
Y=fftshift(fft(A));
N=length(Y);
P_input=Y.*conj(Y);%spectral amplitudes of the pulse
Max_input_pulse = max(P_input);
P_input=P_input/Max_input_pulse;%Normalized Spectral Intensity
f=linspace(-fn,fn,N+1);
f=f(1:N);
f=f+f_0;
shift_points = 800;
subplot(2,2,2);
shift_points:(N/2)+shift_points),':*b');%spectrum of the pulse
semilogy(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P_input((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the pulse
title('The spectrum of the input pulse');
xlabel('Wavelength (nm)');
Omega_0 = 2*pi*((3e+8/1e+15)/850e-9);%unit is rad/femtosecond
Alpha = 0*(10e-3)/4.343 ;%unit is 1/m
gamma = 0.0452e+15;%1/(W*m)----------1W = 1J/s =1e-15J/fs; fs is femtosecond
Beta_2 = -1.276e+4;%fs^2/m
Beta_3 = 8.119e+4;%fs^3/m
Beta_4 = -1.321e+5;%fs^4/m
Beta_5 = 3.032e+5;%fs^5/m
Beta_6 = -4.196e+5;%fs^6/m
Beta_7 = 2.570e+5;%fs^7/m
```

```
%*****************************************************************
fr=0.18;
tt1=12.2;%femtosecond
tt2=32;%femtosecond
hr_t=((tt1^2+tt2^2)/(tt1*tt2^2))*((exp(-(t/tt2))).*sin(t/tt1));
hr_f = fft(hr_t);
%*****************************************************************
part_A_srs_0 = 0;
part_A_steepening_0 = 0;
part_A_srs_h = 0;
part_A_steepening_h = 0;
%*****************************************************************
h_0 = 0.002e-2;%the lengeh of one step in split step Fourier Method,unit is meter
step_counter = 0;
for I = 1 : Steps;
   if step_counter == 4;
     h = -2*h_0;
     step_counter = -1;
   else
     h = h_0;
   end
 Y = fft(A);
N = length(Y);
Omega = linspace(0,fn,(N/2)+1);
Omega_1 = 2*pi*(1e-15)*Omega(1:N/2);%unit is rad/femtosecond
Omega_2 = -2*pi*(1e-15)*linspace(fn,Omega(2),N/2);%unit is rad/femtosecond
Omega(1:N/2) = Omega_1;
Omega(((N/2)+1):N) = Omega_2;
D_f= - j*(Beta_2/2)*(j*Omega).^2 + (Beta_3/6)*(j*Omega).^3 +
j*(Beta_4/24)*(j*Omega).^4 - (Beta_5/120)*(j*Omega).^5 -
j*(Beta_6/720)*(j*Omega).^6 + (Beta_7/5040)*(j*Omega).^7 - Alpha/2 ;%frequency
domain
Y = exp(0.5*h*D_f).*Y;
%P=Y(1:N/2).*conj(Y(1:N/2));%spectral amplitudes of the pulse;
A_D1_t=ifft(Y);
%*****************************************************************
A_v = A;
for k = 1:N
   if A(k) == 0;
     A_v(k) = 1e-100;
   else
     A_v(k) = A(k);
   end
end
A_srs_1 = ifft(hr_f.*fft((abs(A)).^2));
Z = A.*A_srs_1;
```

```
dt = ts;
M = N;
sign = part_A_srs_0;
[A_srs_2,part_A_srs_0] = Derivation(Z,dt,M,sign);
A_srs = A_srs_1 + ((j/Omega_0)*A_srs_2)./A_v;
%*****************************************************
Z = ((abs(A)).^2).*A;
dt = ts;
M = N;
sign = part_A_steepening_0;
[A_steepening,part_A_steepening_0] = Derivation(Z,dt,M,sign);
A_steepening = A_steepening./A_v;
A_steepening = j*A_steepening/Omega_0;
%*****************************************************
A_spm = (abs(A)).^2;
%*****************************************************
N_0 = j*gamma*((1-fr)*(A_spm + A_steepening) + fr*A_srs);
%*****************************************************
A_N = (exp(0.5*h*N_0 + 0.5*h*N_0)).*A_D1_t;
A_N_f = fft(A_N);
A_D2_f = (exp(0.5*h*D_f)).*A_N_f;
A_h_t = ifft(A_D2_f);
A = A_h_t;
%*****************************************************
A_v = A;
for k = 1:N
   if A(k) == 0;
     A_v(k) = 1e-100;
   else
     A_v(k) = A(k);
   end
end
%*****************************************************
A_srs_1 = ifft(hr_f.*fft((abs(A)).^2));
Z = A.*A_srs_1;
dt = ts;
M = N;
sign = part_A_srs_h;
[A_srs_2,part_A_srs_h] = Derivation(Z,dt,M,sign);
A_srs = A_srs_1 + ((j/Omega_0)*A_srs_2)./A_v;
%*****************************************************
Z = ((abs(A)).^2).*A;
dt = ts;
M = N;
sign = part_A_steepening_h;
[A_steepening,part_A_steepening_h] = Derivation(Z,dt,M,sign);
```

```
A_steepening = A_steepening./A_v;
A_steepening = j*A_steepening/Omega_0;
A_spm = (abs(A)).^2;
%*****************************************************
N_h = j*gamma*((1-fr)*(A_spm + A_steepening) + fr*A_srs);
%*****************************************************
A_N = (exp(0.5*h*N_0 + 0.5*h*N_h)).*A_D1_t;
A_N_f = fft(A_N);
A_D2_f = (exp(0.5*h*D_f)).*A_N_f;
A_h_t = ifft(A_D2_f);
A = A_h_t;
step_counter = step_counter + 1;
end
subplot (2,2,3);
Output_pulse = (abs(A)).^2;
Max_output_pulse = max((abs(A).^2));
Output_pulse = Output_pulse/Max_output_pulse;%Normalized output pulse
plot(t,Output_pulse,'r');
title('Output pulse in time domain');
xlabel('Time (femtosecond)');
grid on;
Y=fft(A);
Y=fftshift(Y);
P=Y.*conj(Y);%spectral amplitudes of the pulse
Max_output_spectrum = max(P);
P=P/Max_output_spectrum;%Normalized Spectral Intensity
subplot(2,2,4);
semilogy(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
title('The specturm of the output pulse');
xlabel('Wavelength (nm)');
figure;
semilogy(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P_input((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the pulse
title('The spectrum of the input pulse');
xlabel('Wavelengeh (nm)');
figure;
semilogy(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
title('The spectrum of the output pulse');
xlabel('Wavelength (nm)');
etime(clock,ti)
clear;
```

Appendix II

```
clc;%clear command window
clear;
close;
Steps =10000;
P_0 = 10000;%unit W = J/S= J/fs*1e+15
ti = clock;
Lambda_0 = 850;%unit is nm(central frequency-carrier frequency)
f_0=3e+8/(Lambda_0*1e-9);%central frequency(Hz)
ts=1;%1 femtosecond,sampling period.
fn=1/(2*ts*1e-15);%Nyquist frequency(Hz)
t = 0:ts:2999;%unit is femtosecond
shift=500;%unit is femtosecond
T0=56.7;%femtosecond,T0 width of the pulse
A=sqrt(P_0/1e+15)*(sech((t-shift)/T0));%Input pulse A(0,t)-(Hyperbolic pulse)
subplot(2,2,1);
plot(t,((abs(A)).^2)/(P_0/1e+15));%amplitudes of the pulse in time domain
title('Input pulse in time domain');
xlabel('Time (femtosecond)');
grid on;
Y=fftshift(fft(A));
N=length(Y);
P_input=Y.*conj(Y);%spectral amplitudes of the pulse
Max_input_pulse = max(P_input);
P_input=P_input/Max_input_pulse;%Normalized Spectral Intensity
f=linspace(-fn,fn,N+1);
f=f(1:N);
f=f+f_0;
shift_points = 800;
subplot(2,2,2);
%plot(3e+17./f((N/2)-shift_points:(N/2)+shift_points),P_input((N/2)-
shift_points:(N/2)+shift_points),':*b');%spectrum of the pulse
semilogy(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P_input((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the pulse
title('The spectrum of the input pulse');
xlabel('Wavelengeh (nm)');
Omega_0 = 2*pi*((3e+8/1e+15)/850e-9);%unit is rad/femtosecond
Alpha = 0*(10e-3)/4.343 ;%unit is 1/m
gamma = 0.0452e+15;%1/(W*m)----------1W = 1J/s =1e-15J/fs; fs is femtosecond
Beta_2 = -1.276e+4;%fs^2/m
Beta_3 = 8.119e+4;%fs^3/m
Beta_4 = -1.321e+5;%fs^4/m
Beta_5 = 3.032e+5;%fs^5/m
Beta_6 = -4.196e+5;%fs^6/m
```

```
Beta_7 = 2.570e+5;%fs^7/m
N_0 = 0;
N_1 = 0;
N_2 = 0;
N_3 = 0;
N_4 = 0;
a_0 = 1901/720;
a_1 = -2774/720;
a_2 = 2616/720;
a_3 = -1274/720;
a_4 = 251/720;
c1 = 251/1440;
c_0 = 897/1440;
c_1 = 382/1440;
c_2 = -158/1440;
c_3 = 87/1440;
c_4 = -19/1440;
%*************************************************************************
****
fr=0.18;
tt1=12.2;%femtosecond
tt2=32;%femtosecond
hr_t=((tt1^2+tt2^2)/(tt1*tt2^2))*((exp(-(t/tt2))).*sin(t/tt1));
hr_f = fft(hr_t);
%*************************************************************************
part_A_srs_0 = 0;
part_A_steepening_0 = 0;
part_A_srs_h = 0;
part_A_steepening_h = 0;
%*************************************************************************
h = 0.0025e-2;%the lengeh of one step in split step Fourier Method,unit is meter
for I = 1 : Steps;
Y = fft(A);
N = length(Y);
Omega = linspace(0,fn,(N/2)+1);
Omega_1 = 2*pi*(1e-15)*Omega(1:N/2);%unit is rad/femtosecond
Omega_2 = -2*pi*(1e-15)*linspace(fn,Omega(2),N/2);%unit is rad/femtosecond
Omega(1:N/2) = Omega_1;
Omega(((N/2)+1):N) = Omega_2;
D_f= - j*(Beta_2/2)*(j*Omega).^2 + (Beta_3/6)*(j*Omega).^3 +
j*(Beta_4/24)*(j*Omega).^4 - (Beta_5/120)*(j*Omega).^5 -
j*(Beta_6/720)*(j*Omega).^6 + (Beta_7/5040)*(j*Omega).^7 - Alpha/2 ;%frequency
domain
Y = exp(0.5*h*D_f).*Y;
%P=Y(1:N/2).*conj(Y(1:N/2));%spectral amplitudes of the pulse;
A_D1_t=ifft(Y);
```

```
%*****************************************************************
A_v = A;
for k = 1:N
   if A(k) == 0;
     A_v(k) = 1e-100;
   else
     A_v(k) = A(k);
   end
end
%*****************************************************************
%This is the step to acquire N(0)
%*****************************************************************
A_srs_1 = ifft(hr_f.*fft((abs(A)).^2));
Z = A.*A_srs_1;
dt = ts;
M = N;
sign = part_A_srs_0;
[A_srs_2,part_A_srs_0] = Derivation(Z,dt,M,sign);
A_srs = A_srs_1 + ((j/Omega_0)*A_srs_2)./A_v;
%*****************************************************
Z = ((abs(A)).^2).*A;
dt = ts;
M = N;
sign = part_A_steepening_0;
[A_steepening,part_A_steepening_0] = Derivation(Z,dt,M,sign);
A_steepening = A_steepening./A_v;
A_steepening = j*A_steepening/Omega_0;
%*****************************************************
A_spm = (abs(A)).^2;
%*****************************************************
N_0 = j*gamma*((1-fr)*(A_spm + A_steepening) + fr*A_srs);
%*****************************************************
A_N = ( exp ( h*( a_0 * N_0 + a_1 * N_1 + a_2 * N_2 + a_3 * N_3 + a_4 *
N_4 ))).*A_D1_t;
A_N_f = fft(A_N);
A_D2_f = (exp(0.5*h*D_f)).*A_N_f;
A_h_t = ifft(A_D2_f);
A = A_h_t;
%*****************************************************
A_v = A;
for k = 1:N
   if A(k) == 0;
     A_v(k) = 1e-100;
   else
     A_v(k) = A(k);
   end
```

```
end
%*****************************************************
A_srs_1 = ifft(hr_f.*fft((abs(A)).^2));
Z = A.*A_srs_1;
dt = ts;
M = N;
sign = part_A_srs_h;
[A_srs_2,part_A_srs_h] = Derivation(Z,dt,M,sign);
A_srs = A_srs_1 + ((j/Omega_0)*A_srs_2)./A_v;
%*****************************************************
Z = ((abs(A)).^2).*A;
dt = ts;
M = N;
sign = part_A_steepening_h;
[A_steepening,part_A_steepening_h] = Derivation(Z,dt,M,sign);
A_steepening = A_steepening./A_v;
A_steepening = j*A_steepening/Omega_0;
%*****************************************************
A_spm = (abs(A)).^2;
%*****************************************************
N1 = j*gamma*((1-fr)*(A_spm + A_steepening) + fr*A_srs);
%*****************************************************
A_N = ( exp ( h*( c1 * N1 + c_0 * N_0 + c_1 * N_1 + c_2 * N_2 + c_3 * N_3 + c_4 *
N_4 ))).*A_D1_t;
N_4 = N_3;
N_3 = N_2;
N_2 = N_1;
N_1 = N_0;
A_N_f = fft(A_N);
A_D2_f = (exp(0.5*h*D_f)).*A_N_f;
A_h_t = ifft(A_D2_f);
A = A_h_t;
end
subplot (2,2,3);
Output_pulse = (abs(A)).^2;
Max_output_pulse = max((abs(A).^2));
Output_pulse = Output_pulse/Max_output_pulse;%Normalized output pulse
plot(t,Output_pulse,'r');
title('Output pulse in time domain');
xlabel('Time (femtosecond)');
grid on;
Y=fft(A);
Y=fftshift(Y);
P=Y.*conj(Y);%spectral amplitudes of the pulse
Max_output_spectrum = max(P);
P=P/Max_output_spectrum;%Normalized Spectral Intensity
```

```
subplot(2,2,4);
semilogy(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
title('The spectrum of the output pulse');
xlabel('Wavelength (nm)');
%*****************************************************
figure;
semilogy(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P_input((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the pulse
title('The spectrum of the input pulse');
xlabel('Wavelengeh (nm)');
figure;
semilogy(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
title('The spectrum of the output pulse');
xlabel('Wavelength (nm)');
%*****************************************************
etime(clock,ti)
clear;
```

Appendix III

```
clc;%clear command window
clear;
close;
Steps =15000;
P_0 = 10000;%unit W = J/S= J/fs*1e+15
ti = clock;
Lambda_0 = 850;%unit is nm(central frequency-carrier frequency)
f_0=3e+8/(Lambda_0*1e-9);%central frequency(Hz)
ts=1;%1 femtosecond,sampling period.
fn=1/(2*ts*1e-15);%Nyquist frequency(Hz)
t = 0:ts:2999;%unit is femtosecond
shift=500;%unit is femtosecond
T0=57;%femtosecond,T0 width of the pulse
A=sqrt(P_0/1e+15)*sech((t-shift)/T0);%Input pulse A(0,t)-(Hyperbolic pulse)
subplot(2,2,1);
plot(t,((abs(A)).^2)/(P_0/1e+15));%amplitudes of the pulse in time domain
title('Input pulse in time domain');
xlabel('Time (femosecond)');
grid on;
Y=fftshift(fft(A));
N=length(Y);
P_input=Y.*conj(Y);%spectral amplitudes of the pulse
Max_input_pulse = max(P_input);
P_input=P_input/Max_input_pulse;%Normalized Spectral Intensity
f=linspace(-fn,fn,N+1);
f=f(1:N);
f=f+f_0;
shift_points = 800;
subplot(2,2,2);
plot(3e+17./f((N/2)-shift_points:(N/2)+shift_points),P_input((N/2)-
shift_points:(N/2)+shift_points),':*b');%spectrum of the pulse
title('The spectrum of the input pulse');
xlabel('Wavelength (nm)');
Omega_0 = 2*pi*((3e+8/1e+15)/850e-9);%unit is rad/femtosecond
Alpha = 0*(10e-3)/4.343 ;%unit is 1/m
gamma = 0.0452e+15;%1/(W*m)----------1W = 1J/s =1e-15J/fs; fs is femtosecond
Beta_2 = -1.276e+4;%fs^2/m
Beta_3 = 8.119e+4;%fs^3/m
Beta_4 = -1.321e+5;%fs^4/m
Beta_5 = 3.032e+5;%fs^5/m
Beta_6 = -4.196e+5;%fs^6/m
Beta_7 = 2.570e+5;%fs^7/m
%*****************************************************************
```

```
fr=0.18;
tt1=12.2;%femtosecond
tt2=32;%femtosecond
hr_t=((tt1^2+tt2^2)/(tt1*tt2^2))*((exp(-(t/tt2))).*sin(t/tt1));
hr_f = fft(hr_t);
%*********************************************************************
part_A_srs_0 = 0;
part_A_steepening_0 = 0;
part_A_srs_h = 0;
part_A_steepening_h = 0;
%*********************************************************************
h_0 = 0.001e-2;%the lengeh of one step in split step Fourier Method,unit is meter
step_counter = 0;
for I = 1 : Steps;
   if step_counter == 4;
     h = -2*h_0;
     step_counter = -1;
   else
     h = h_0;
   end

Y = fft(A);
N = length(Y);
Omega = linspace(0,fn,(N/2)+1);
Omega_1 = 2*pi*(1e-15)*Omega(1:N/2);%unit is rad/femtosecond
Omega_2 = -2*pi*(1e-15)*linspace(fn,Omega(2),N/2);%unit is rad/femtosecond
Omega(1:N/2) = Omega_1;
Omega((N/2)+1):N) = Omega_2;
D_f= - j*(Beta_2/2)*(j*Omega).^2 + (Beta_3/6)*(j*Omega).^3 +
j*(Beta_4/24)*(j*Omega).^4 - (Beta_5/120)*(j*Omega).^5 -
j*(Beta_6/720)*(j*Omega).^6 + (Beta_7/5040)*(j*Omega).^7 - Alpha/2 ;%frequency
domain
Y = exp(0.5*h*D_f).*Y;
%P=Y(1:N/2).*conj(Y(1:N/2));%spectral amplitudes of the pulse;
A_D1_t=ifft(Y);
%*********************************************************************
A_v = A;
for k = 1:N
   if A(k) == 0;
     A_v(k) = 1e-100;
   else
     A_v(k) = A(k);
   end
end
%*********************************************************************
A_srs_1 = ifft(hr_f.*fft((abs(A)).^2));
```

```
Z = A.*A_srs_1;
dt = ts;
M = N;
sign = part_A_srs_0;
[A_srs_2,part_A_srs_0] = Derivation(Z,dt,M,sign);

A_srs = A_srs_1 + ((j/Omega_0)*A_srs_2)./A_v;
%*******************************************************
Z = ((abs(A)).^2).*A;
dt = ts;
M = N;
sign = part_A_steepening_0;
[A_steepening,part_A_steepening_0] = Derivation(Z,dt,M,sign);
A_steepening = A_steepening./A_v;
A_steepening = j*A_steepening/Omega_0;
%*******************************************************
A_spm = (abs(A)).^2;
%*******************************************************
N_0 = j*gamma*((1-fr)*(A_spm + A_steepening) + fr*A_srs);
%*******************************************************
A_N = (exp(0.5*h*N_0 + 0.5*h*N_0)).*A_D1_t;
A_N_f = fft(A_N);
A_D2_f = (exp(0.5*h*D_f)).*A_N_f;
A_h_t = ifft(A_D2_f);
A = A_h_t;
%*******************************************************
A_v = A;
for k = 1:N
   if A(k) == 0;
     A_v(k) = 1e-100;
   else
     A_v(k) = A(k);
   end
end
%*******************************************************
A_srs_1 = ifft(hr_f.*fft((abs(A)).^2));
Z = A.*A_srs_1;
dt = ts;
M = N;
sign = part_A_srs_h;
[A_srs_2,part_A_srs_h] = Derivation(Z,dt,M,sign);
A_srs = A_srs_1 + ((j/Omega_0)*A_srs_2)./A_v;
%*******************************************************
Z = ((abs(A)).^2).*A;
dt = ts;
M = N;
```

```
sign = part_A_steepening_h;
[A_steepening,part_A_steepening_h] = Derivation(Z,dt,M,sign);
A_steepening = A_steepening./A_v;
A_steepening = j*A_steepening/Omega_0;
A_spm = (abs(A)).^2;
N_h = j*gamma*((1-fr)*(A_spm + A_steepening) + fr*A_srs);
A_N = (exp(0.5*h*N_0 + 0.5*h*N_h)).*A_D1_t;
A_N_f = fft(A_N);
A_D2_f = (exp(0.5*h*D_f)).*A_N_f;
A_h_t = ifft(A_D2_f);
A = A_h_t;
step_counter = step_counter + 1;
    if I == 15000,
    B10 = A;
  end
    if I == 13500,
    B9 = A;
  end
    if I == 12000,
    B8 = A;
  end
    if I == 10500,
    B7 = A;
  end
    if I == 9000,
    B6 = A;
  end
    if I == 7500,
    B5 = A;
  end
    if I == 6000,
    B4 = A;
  end
    if I == 4500,
    B3 = A;
  end
    if I == 3000,
    B2 = A;
  end
    if I == 1500,
    B1 = A;
  end
  if I == 500,
    B05 = A;
  end
end
```

```
save datax_spectrum_total_4order B05 B1 B2 B3 B4 B5 B6 B7 B8 B9 B10 Steps h t f N
P_0 shift_points;
subplot (2,2,3);
Output_pulse = (abs(A)).^2;
Max_output_pulse = max((abs(A).^2));
Output_pulse = Output_pulse/Max_output_pulse;%Normalized output pulse
plot(t,Output_pulse,'r');
title('Output pulse in time domain');
xlabel('Time (femosecond)');
grid on;
Y=fft(A);
Y=fftshift(Y);
P=Y.*conj(Y);%spectral amplitudes of the pulse
Max_output_spectrum = max(P);
P=P/Max_output_spectrum;%Normalized Spectral Intensity
subplot(2,2,4);
semilogy(3e+17./f((N/2)-shift_points:(N/2)+shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
title('The specturm of the output pulse');
xlabel('Wavelength (nm)');
figure;
plot(3e+17./f((N/2)-shift_points:(N/2)+shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
title('The specturm of the output pulse');
xlabel('Wavelengeh (nm)');
figure;
semilogy(3e+17./f((N/2)-shift_points:(N/2)+shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
title('The specturm of the output pulse');
xlabel('Wavelength (nm)');
figure;
plot(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the pulse
figure;
semilogy(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the pulse
etime(clock,ti)
clear;
```

Appendix IV

```
clc;%clear command window
clear;
close;
Steps =10000;
P_0 = 10000;%unit W = J/S= J/fs*1e+15
ti = clock;
Lambda_0 = 850;%unit is nm(central frequency-carrier frequency)
f_0=3e+8/(Lambda_0*1e-9);%central frequency(Hz)
ts=1;%1 femtosecond,sampling period.
fn=1/(2*ts*1e-15);%Nyquist frequency(Hz)
t = 0:ts:2999;%unit is femtosecond
shift=500;%unit is femtosecond
T0=56;%femtosecond,T0 width of the pulse
A=sqrt(P_0/1e+15)*(sech((t-shift)/T0));%Input pulse A(0,t)-(Hyperbolic pulse)
subplot(2,2,1);
plot(t,((abs(A)).^2)/(P_0/1e+15));%amplitudes of the pulse in time domain
title('Input pulse in time domain');
xlabel('Time (femosecond)');
grid on;
Y=fftshift(fft(A));
N=length(Y);
P_input=Y.*conj(Y);%spectral amplitudes of the pulse
Max_input_pulse = max(P_input);
P_input=P_input/Max_input_pulse;%Normalized Spectral Intensity
f=linspace(-fn,fn,N+1);
f=f(1:N);
f=f+f_0;
shift_points = 800;
subplot(2,2,2);
plot(3e+17./f((N/2)-shift_points:(N/2)+shift_points),P_input((N/2)-
shift_points:(N/2)+shift_points),':*b');%spectrum of the pulse
title('The specturm of the input pulse');
xlabel('Wavelengeh (nm)');
Omega_0 = 2*pi*((3e+8/1e+15)/850e-9);%unit is rad/femtosecond
Alpha = 0*(10e-3)/4.343 ;%unit is 1/m
gamma = 0.0452e+15;%1/(W*m)----------1W = 1J/s =1e-15J/fs; fs is femtosecond
Beta_2 = -1.276e+4;%fs^2/m
Beta_3 = 8.119e+4;%fs^3/m
Beta_4 = -1.321e+5;%fs^4/m
Beta_5 = 3.032e+5;%fs^5/m
Beta_6 = -4.196e+5;%fs^6/m
Beta_7 = 2.570e+5;%fs^7/m
N_0 = 0;
```

```
N_1 = 0;
N_2 = 0;
N_3 = 0;
N_4 = 0;
a_0 = 1901/720;
a_1 = -2774/720;
a_2 = 2616/720;
a_3 = -1274/720;
a_4 = 251/720;
c1 = 251/1440;
c_0 = 897/1440;
c_1 = 382/1440;
c_2 = -158/1440;
c_3 = 87/1440;
c_4 = -19/1440;
fr=0.18;
tt1=12.2;%femtosecond
tt2=32;%femtosecond
hr_t=((tt1^2+tt2^2)/(tt1*tt2^2))*((exp(-(t/tt2))).*sin(t/tt1));
hr_f = fft(hr_t);
part_A_srs_0 = 0;
part_A_steepening_0 = 0;
part_A_srs_h = 0;
part_A_steepening_h = 0;
h = 0.001e-2;%the lengeh of one step in split step Fourier Method,unit is meter
for I = 1 : Steps;
Y = fft(A);
N = length(Y);
Omega = linspace(0,fn,(N/2)+1);
Omega_1 = 2*pi*(1e-15)*Omega(1:N/2);%unit is rad/femtosecond
Omega_2 = -2*pi*(1e-15)*linspace(fn,Omega(2),N/2);%unit is rad/femtosecond
Omega(1:N/2) = Omega_1;
Omega(((N/2)+1):N) = Omega_2;
D_f= - j*(Beta_2/2)*(j*Omega).^2 + (Beta_3/6)*(j*Omega).^3 +
j*(Beta_4/24)*(j*Omega).^4 - (Beta_5/120)*(j*Omega).^5 -
j*(Beta_6/720)*(j*Omega).^6 + (Beta_7/5040)*(j*Omega).^7 - Alpha/2 ;%frequency
domain
Y = exp(0.5*h*D_f).*Y;
%P=Y(1:N/2).*conj(Y(1:N/2));%spectral amplitudes of the pulse;
A_D1_t=ifft(Y);
%**************************************************************
A_v = A;
for k = 1:N
   if A(k) == 0;
     A_v(k) = 1e-100;
   else
```

```
    A_v(k) = A(k);
  end
end
A_srs_1 = ifft(hr_f.*fft((abs(A)).^2));
Z = A.*A_srs_1;
dt = ts;
M = N;
sign = part_A_srs_0;
[A_srs_2,part_A_srs_0] = Derivation(Z,dt,M,sign);
A_srs = A_srs_1 + ((j/Omega_0)*A_srs_2)./A_v;
%*******************************************************
Z = ((abs(A)).^2).*A;
dt = ts;
M = N;
sign = part_A_steepening_0;
[A_steepening,part_A_steepening_0] = Derivation(Z,dt,M,sign);
A_steepening = A_steepening./A_v;
A_steepening = j*A_steepening/Omega_0;
%*******************************************************
A_spm = (abs(A)).^2;
%*******************************************************
N_0 = j*gamma*((1-fr)*(A_spm + A_steepening) + fr*A_srs);
%*******************************************************
A_N = ( exp ( h*( a_0 * N_0 + a_1 * N_1 + a_2 * N_2 + a_3 * N_3 + a_4 *
N_4 ))).*A_D1_t;
A_N_f = fft(A_N);
A_D2_f = (exp(0.5*h*D_f)).*A_N_f;
A_h_t = ifft(A_D2_f);
A = A_h_t;
%*******************************************************
A_v = A;
for k = 1:N
  if A(k) == 0;
    A_v(k) = 1e-100;
  else
    A_v(k) = A(k);
  end
end
%*******************************************************
A_srs_1 = ifft(hr_f.*fft((abs(A)).^2));
Z = A.*A_srs_1;
dt = ts;
M = N;
sign = part_A_srs_h;
[A_srs_2,part_A_srs_h] = Derivation(Z,dt,M,sign);
A_srs = A_srs_1 + ((j/Omega_0)*A_srs_2)./A_v;
```

```
%*****************************************************
Z = ((abs(A)).^2).*A;
dt = ts;
M = N;
sign = part_A_steepening_h;
[A_steepening,part_A_steepening_h] = Derivation(Z,dt,M,sign);
A_steepening = A_steepening./A_v;
A_steepening = j*A_steepening/Omega_0;
%*****************************************************
A_spm = (abs(A)).^2;
%*****************************************************
N1 = j*gamma*((1-fr)*(A_spm + A_steepening) + fr*A_srs);
%*****************************************************
A_N = ( exp ( h*( c1 * N1 + c_0 * N_0 + c_1 * N_1 + c_2 * N_2 + c_3 * N_3 + c_4 *
N_4 ))).*A_D1_t;
N_4 = N_3;
N_3 = N_2;
N_2 = N_1;
N_1 = N_0;
A_N_f = fft(A_N);
A_D2_f = (exp(0.5*h*D_f)).*A_N_f;
A_h_t = ifft(A_D2_f);
A = A_h_t;
    if I == 10000,
      B10 = A;
   end
      if I == 9000,
      B9 = A;
   end
      if I == 8000,
      B8 = A;
   end
      if I == 7000,
      B7 = A;
   end
      if I == 6000,
      B6 = A;
   end
      if I == 5000,
      B5 = A;
   end
      if I == 4000,
      B4 = A;
   end
      if I == 3000,
      B3 = A;
```

```
  end
  if I == 2000,
    B2 = A;
  end
  if I == 1000,
    B1 = A;
    etime(clock,ti)
  end
  if I == 500,
    B05 = A;
  end
end
save datax_spectrum_total_pc B05 B1 B2 B3 B4 B5 B6 B7 B8 B9 B10 Steps h t f N P_0
shift_points;
subplot (2,2,3);
Output_pulse = (abs(A)).^2;
Max_output_pulse = max((abs(A).^2));
Output_pulse = Output_pulse/Max_output_pulse;%Normalized output pulse
plot(t,Output_pulse,'r');
title('Output pulse in time domain');
xlabel('Time (femosecond)');
grid on;
Y=fft(A);
Y=fftshift(Y);
P=Y.*conj(Y);%spectral amplitudes of the pulse
Max_output_spectrum = max(P);
P=P/Max_output_spectrum;%Normalized Spectral Intensity
subplot(2,2,4);
semilogy(3e+17./f((N/2)-shift_points:(N/2)+shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
title('The specturm of the output pulse');
xlabel('Wavelength (nm)');
figure;
plot(3e+17./f((N/2)-shift_points:(N/2)+shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
title('The specturm of the output pulse');
xlabel('Wavelengeh (nm)');
figure;
semilogy(3e+17./f((N/2)-shift_points:(N/2)+shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
title('The specturm of the output pulse');
xlabel('Wavelength (nm)');
figure;
plot(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
figure;
```

```
semilogy(3e+17./f((N/2)+shift_points:-1:(N/2)-shift_points),P((N/2)-
shift_points:(N/2)+shift_points),'b');%spectrum of the output pulse
etime(clock,ti)
%*************************************************
clear;
```