

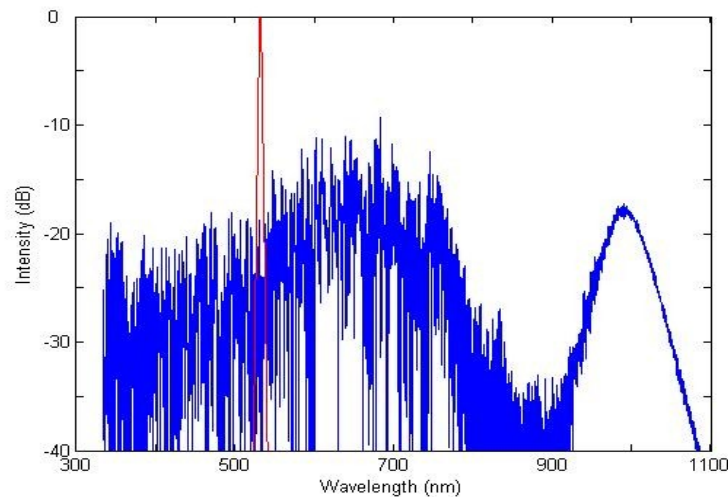
# Contents

1. Summary.....	2
2. Introduction.....	3
3. Background Theory.....	3
3.1. Generalised Nonlinear Schrödinger Equation.....	3
3.2. Fibre Losses.....	4
3.3. Nonlinear Response Function.....	4
3.4. Nonlinear Coefficients.....	5
3.5. Simplifying the Generalised Nonlinear Schrödinger Equation.....	5
3.6. Split Step Fourier Method for Solving the NLSE.....	6
4. Numerically Simulating the Symmetrised SSFM.....	8
4.1. Implementing the SSFM.....	8
4.2. Numerical Simulation.....	8
4.3. Choosing the Step Size and Time Window.....	9
4.4. Order of the Simulation Program.....	10
5. Testing the Numerical Simulation Program.....	10
5.1. Optical Solitons.....	11
5.2. Raman Scattering of Solitons.....	13
5.3. Comparison of Numerical Simulation with Paper Cases.....	14
6. Simulating Supercontinuum Generation in the Ultraviolet End of the Spectrum.....	16
6.1. Estimating Fibre Parameters.....	16
6.2. Estimated Beta Values for Different ZDWs and Laser Wavelengths.....	17
6.3. Sensitivity Analysis of the Simulations to Changes in Variables.....	18
6.3.1. Base Case and Variable Changes for an 800 nm Laser.....	18
6.3.2. Base Case and Variable Changes for a 532 nm Laser.....	22
6.4. Simulating Broadening in the Ultraviolet Region of the Electromagnetic Spectrum.....	24
7. Discussion.....	26
8. Proposals for Further Work.....	26
9. Conclusions.....	27
Appendix 1: Matlab™ Code.....	27
Appendix 1.1: Main Code Used By A Typical Example.....	27
Appendix 1.2: Code for the Function SSFM.....	30
Appendix 1.3: Other Functions Used By The Main Code & SSFM.....	32
Appendix 2: Fourier Transforms.....	32
Appendix 3: Nomenclature.....	33

## 1. Summary

This project aims to numerically simulate the formation of a supercontinuum in the near-ultraviolet (around 200 – 400 nm) region of the electromagnetic spectrum. In order to do this a numerical model to solve the nonlinear Schrödinger equation governing supercontinuum generation was written using Matlab software (shown in Appendix 1).

The program was then tested against examples from the scientific literature. An analysis of how changes in the variables and the fibre parameters affected the supercontinua that were generated was undertaken. The changes that produced the greatest amount of near-ultraviolet spectral broadening were then combined and used to produce as broad a spectrum as possible. Figure 1 shows the supercontinuum that gave good near-ultraviolet broadening.



**Figure 1:** *Supercontinuum (blue line) generated after propagating a 532 nm laser pulse (red line) down a fibre of length 0.3 m.*

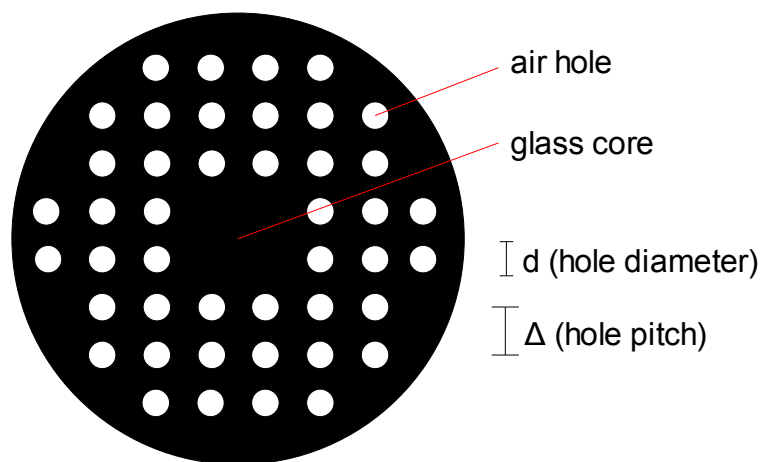
This project was undertaken under Dr. J. Hult of the Laser Analytics Group in the Department of Chemical Engineering of Cambridge University. The book “Nonlinear Fibre Optics” (Agrawal, 2007) contains a lot of the background theory used by this project. The Split-Step Propagation (SSPROP) Matlab code (Murphy, 2006) formed the basis of the numerical algorithm used in the numerical simulation.

Ultraviolet supercontinua could be used in metrology and imaging, because supercontinua have a higher intensity and coherence than conventional light sources and have a broader spectrum than laser light. Fibres that produce good ultraviolet supercontinua should be available in the near future and this project aims to predict how they would look.

## 2. Introduction

A supercontinuum is an optical pulse with a broad spectrum that can extend over hundreds of nanometres. Supercontinua can be generated by sending very short, intense (high power) optical pulses into photonic crystal fibres (PCFs). These pulses are of the order of a few femtoseconds long. In order to simulate the formation of supercontinua, an understanding of the media in which they are generated is needed. Optical fibres channel light by using total internal reflection, which occurs when a large difference in material refractive indices exists. PCFs create this difference by using a matrix of materials with different refractive indices.

Most PCFs used for SC generation comprise of a solid silica core surrounded by a matrix of silica and air. The solid core has a higher refractive index than the cladding matrix. The air holes allow more control over the wave guidance and enable a shift of the zero-dispersion wavelength (ZDW) from 1.3  $\mu\text{m}$  (the value for standard solid silica fibres) to lower values around 800 nm (Hansen & Kristiansen, 2007). The ZDW is the wavelength at which dispersion due to material effects is zero (for a single mode fibre), enabling nonlinear effects to become more significant, which is important for supercontinuum generation.



**Figure 2:** *Cross-section of a common index-guiding photonic crystal fibre.*

### 3. Background Theory

#### 3.1. Generalised Nonlinear Schrödinger Equation

Equation (1), the Generalised Nonlinear Schrödinger Equation (GNLSE), is to be solved for  $A(z,t)$ , the field envelope, and is taken from Agrawal (2007), Kobtsev & Smirnov (2006) and Dudley et al. (2006).

$$\frac{\partial A}{\partial z} = -\frac{\alpha}{2} - \sum_{m=2}^M i^{m-1} \frac{\beta_m}{m!} \frac{\partial^m A}{\partial t^m} + i \left( \gamma + i \gamma_1 \frac{\partial}{\partial t} \right) \left( A(z,t) \int_0^\infty R(t') |A(z, t-t')|^2 dt' \right) \quad (1)$$

#### 3.2. Fibre Losses

$\alpha$  in Equation (1) represents the power losses in the fibre as the light travels through it. For supercontinuum generation, the length of fibre used is small (only a few metres), so these losses can be neglected in the numerical simulation.

#### 3.3. Nonlinear Response Function

$R(t')$  is the nonlinear response function and takes account of the electronic and nuclear contributions and takes the form of Equation (2).

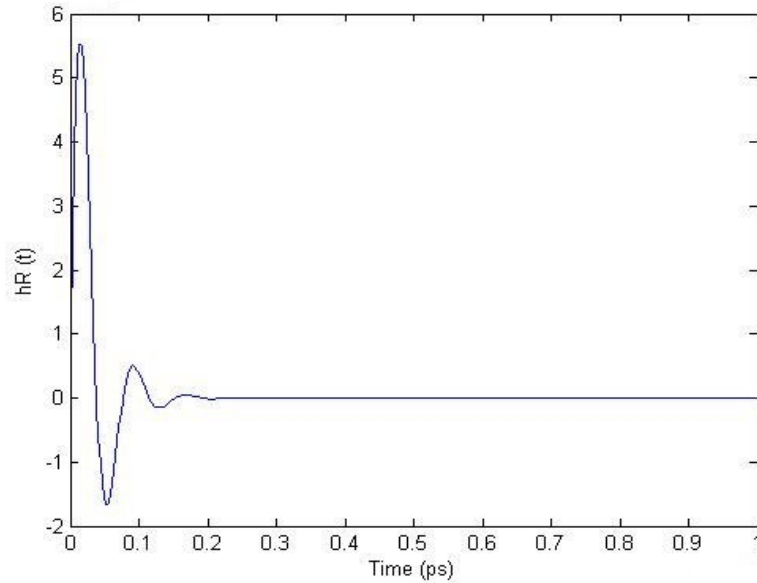
$$R(t') = (1 - f_R) \delta(t - t_e) + f_R h_R(t) \quad (2)$$

The electronic contribution is treated in this analysis as occurring instantaneously because  $t_e \approx 1$  fs.  $h_R(t)$  is the Raman response function and contains information on the vibration of silica molecules (within the optical fibre) as light passes through the fibre. This is best calculated experimentally, but an analytical form of  $h_R(t)$  exists (Blow & Wood, 1989), given by Equation (3).

Experiments show that  $f_R = 0.18$ ,  $\tau_1 = 12.2$  fs and  $\tau_2 = 32$  fs from Blow & Wood (1989).

$$h_R(t) = \frac{\tau_1^2 + \tau_2^2}{\tau_1 \tau_2^2} e^{\left(\frac{-t}{\tau_2}\right)} \sin\left(\frac{t}{\tau_1}\right) \quad (3)$$

Figure 3 is a graphical representation of Equation (3). Compared with Figure 2.2 from Agrawal (2007), which is experimentally determined, the theoretical model matches well.



**Figure 3:** *Variation of the Raman response function with time.*

### 3.4. Nonlinear Coefficients

The approximation  $\gamma_1 = \frac{\gamma}{\omega_0}$  is made here. This can be used for broadening below 20 THz (Agrawal, 2007). As the wavelength range of supercontinua in this project is from 300 nm (1000 THz) to 2000 nm (150 THz), the frequency range is  $\sim 1000$  THz so this approximation is questionable. A better model of  $\gamma_1$  would be to include the frequency dependence of the effective mode area as described in Dudley et al. (2006). It is shown by the same author in Kibler et al. (2005) that the frequency dependence of the effective mode area is not significant when operating at wavelengths around 800 nm, so for simplicity the above approximation will be kept. It must be noted that for wavelengths over around 1500 nm that the frequency dependence would be significant, so results in that region may be affected.

### 3.5. Simplifying the Generalised Nonlinear Schrödinger Equation

Incorporating the approximations  $\alpha = 0$  and  $\gamma_1 = \frac{\gamma}{\omega_0}$  into Equation (1) yields;

$$\frac{\partial A}{\partial z} = -\sum_{m=2}^M i^{m-1} \frac{\beta_m}{m!} \frac{\partial^m A}{\partial t^m} + i\gamma \left(1 + \frac{i}{\omega_0} \frac{\partial}{\partial t}\right) \left( A(z, t) \int_0^\infty R(t') |A(z, t-t')|^2 dt' \right) \quad (4)$$

Substituting Equation (2) into the integral of Equation (4) gives;

$$\begin{aligned} \int_0^\infty R(t') |A(z, t-t')|^2 dt' &= (1-f_R) \int_0^\infty \delta(t-t_e) |A(z, t-t')|^2 dt' + f_R \int_0^\infty h_R |A(z, t-t')|^2 dt' \\ \rightarrow \int_0^\infty R(t') |A(z, t-t')|^2 dt' &= (1-f_R) |A|^2 + f_R \int_0^\infty h_R |A(z, t-t')|^2 dt' \end{aligned}$$

So Equation (4) becomes;

$$\begin{aligned} \frac{\partial A}{\partial z} &= -\sum_{m=2}^M i^{m-1} \frac{\beta_m}{m!} \frac{\partial^m A}{\partial t^m} + i\gamma A \left(1 + \frac{i}{\omega_0} \frac{\partial}{\partial t}\right) \left( (1-f_R) |A|^2 + f_R \int_0^\infty h_R |A(z, t-t')|^2 dt' \right) \\ \rightarrow \frac{\partial A}{\partial z} &= -\sum_{m=2}^M i^{m-1} \frac{\beta_m}{m!} \frac{\partial^m A}{\partial t^m} + i\gamma (1-f_R) \left( A |A|^2 + \frac{i}{\omega_0} \frac{\partial}{\partial t} (A |A|^2) \right) \\ &+ i\gamma f_R \left( A \int_0^\infty h_R |A(z, t-t')|^2 dt' + \frac{i}{\omega_0} \frac{\partial}{\partial t} \left[ A \int_0^\infty h_R |A(z, t-t')|^2 dt' \right] \right) \end{aligned} \quad (5)$$

### 3.6. Split Step Fourier Method for Solving the NLSE

Equation (5) can now be solved using the SSFM. This method is based around separating the dispersive and nonlinear components of the equation from one another. This assumes that over a very small distance these components may be assumed to be independent. So Equation (5) can be written as:

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

The dispersive terms  $\hat{D}$  and the nonlinear terms  $\hat{N}$  are given by Equation (7) and (8);

$$\hat{D} = -\sum_{m=2}^M i^{m-1} \frac{\beta_m}{m!} \frac{\partial^m}{\partial t^m} \quad (7)$$

$$\begin{aligned} \hat{N} = & i\gamma(1-f_R) \left( |A|^2 + \frac{i}{\omega_0 A} \frac{\partial}{\partial t} (A|A|^2) \right) \\ & + i\gamma f_R \left( \int_0^\infty h_R |A(z, t-t')|^2 dt' + \frac{i}{\omega_0 A} \frac{\partial}{\partial t} \left[ A \int_0^\infty h_R |A(z, t-t')|^2 dt' \right] \right) \end{aligned} \quad (8)$$

Solving Equation (6) gives;

$$A(z+h, t) = \exp[h(\hat{D} + \hat{N})] A(z, t) \quad (9)$$

The Baker-Hausdorff theorem (Weiss & Maradudin, 1962) states that for two non-commuting operators;

$$\exp(h\hat{D}) \exp(h\hat{N}) = \exp\left(h[\hat{D} + \hat{N}] + \frac{h^2}{2}[\hat{D}\hat{N} - \hat{N}\hat{D}] + \dots\right) \quad (10)$$

By ignoring the  $h^2$  and higher terms in the exponential the following approximation can be made;

$$\exp(h[\hat{D} + \hat{N}]) \approx \exp(h\hat{D}) \exp(h\hat{N}) \quad (11)$$

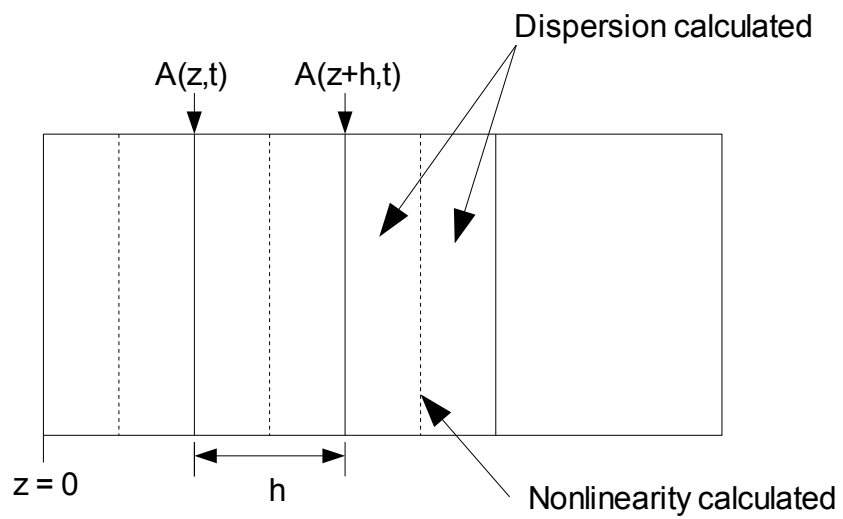
This will result in the SSFM having an accuracy of approximately second order, as the  $h^2$  terms in the Baker-Hausdorff expansion have been neglected. It must be noted that the nonlinear and dispersive terms do not in fact commute. So the solution for the pulse envelope will be;

$$A(z+h, t) \approx \exp(h\hat{D}) \exp(h\hat{N}) A(z, t) \quad (12)$$

A variation on Equation (12) known as the symmetrised SSFM (Fleck et al., 1976) will actually be used in the program, as it has a higher accuracy. It works by calculating the nonlinear effects in the middle of each step as opposed to the edges (see Figure 4). The spatial dependence of the nonlinear term is more accurately described by an integral, whereas in Equation (12) it was approximated by

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

Equation (13) will have an accuracy of approximately third order, as found by applying the Baker-Hausdorff formula twice.



**Figure 4:** Graphical representation of the symmetrised SSFM.



## 4. Numerically Simulating the Symmetrised SSFM

### 4.1. Implementing the SSFM

To simulate supercontinuum generation *Matlab*<sup>TM</sup> was used. Very large numbers of calculations need to be performed and matrices consisting of over 2 million elements are stored. This meant *Mathematica*<sup>TM</sup> could not be used as it cannot cope with such large arrays. Using a language such as C/C++ would have made the simulation even quicker, although *Matlab*<sup>TM</sup> is much more user friendly and allows graphs and data to be created and manipulated more easily.

Some code for the SSFM was freely available (Murphy et al., 2006) and this provided a good starting point for the project, in particular the function 'SSPROP' which executed the symmetrised SSFM. The original code has been added to so that more nonlinear effects are included.

The code used for the simulations is given in Appendix 1. The symmetrised SSFM is performed by the function 'ssfm' which is in Appendix 1.2.

### 4.2. Numerical Simulation

Another advantage of solving the dispersive and nonlinear terms separately is that the dispersive terms can be solved easily by transforming them into the fourier domain. The nonlinear terms cannot and must be solved in the time domain by using a numerical method as they have a spatial dependence. In this project the trapezium rule is used (Agrawal, 2007), which is more accurate than using the basic approximation  $h \hat{N}(z)$  in Equation (12), but others such as the Runge-Kutta methods would have been even more accurate. The integral from Equation (13) becomes;

$$\int_z^{z+h} \hat{N}(z') dz' \approx \frac{h}{2} (\hat{N}(z) + \hat{N}(z+h)) \quad (14)$$

The nonlinearity is calculated at  $(z + h/2)$ , not  $(z + h)$ , so  $\hat{N}(z+h)$  is not known when the integral needs to be approximated in the simulation. To approximate this the following iterative procedure is used;

- 1.) Initially  $\hat{N}(z+h)$  is replaced by  $\hat{N}(z)$  and  $A(z+h,t)$  is calculated.
- 2.) Then  $A(z+h,t)$  is used to calculate a new value for  $\hat{N}(z+h)$ .
- 3.) This new value for  $\hat{N}(z+h)$  is then used to recalculate  $A(z+h,t)$ .

4.) Steps 2) and 3) are repeated until the difference between the previously calculated  $A(z + h, t)$  and the newly calculated value for  $A(z + h, t)$  is within a specified tolerance, or a specified number of iterations have been performed.

Taking the fourier transform of the dispersion terms and the above approximation into account, the program will implement the symmetrised SSFM as follows;

$$A(z+h, t) \approx \mathcal{F}^{-1} \left\{ \exp\left(\frac{h}{2} \hat{D}(i\omega)\right) \mathcal{F} \left\{ \exp(\hat{N}(z) + \hat{N}(z+h)) \mathcal{F}^{-1} \left\{ \exp\left(\frac{h}{2} \hat{D}(i\omega)\right) \mathcal{F} \{A(z, t)\} \right\} \right\} \right\} \quad (15)$$

Fourier transforming  $\hat{D}$  from Equation (7) gives;

$$\mathcal{F} \{\hat{D}\} = \hat{D}(i\omega) = i \sum_{m=2}^M \frac{\beta_m}{m!} (\omega)^m \quad (16)$$

### 4.3. Choosing the Step Size and Time Window

Choosing the appropriate step size and time window for each simulation is important to ensure that the results are accurate and that aliasing (data leaves the window and re-enters at the opposite end) does not occur.

The time window is defined by the multiplication of 'nt' (the number of time steps) and 'dt' (the size of each time step) in the code. Another point to note is that the spectral window is proportional to the inverse of 'dt'. This means that as the time window increases in size, the spectral window will decrease in size. Values of 'nt' and 'dt' that ensure there is no aliasing must be chosen. 'nt' must have values of the form  $2^n$  so that the speed of the fast fourier transforms is maximised. Generally 'nt' takes a value of either  $2^{13}$  or  $2^{14}$  and 'dt' takes values of between 2 and 5 fs.

The step size 'dz' for the integration has to be large enough to ensure that numerical errors do not occur during the simulation, but small enough to ensure that the computational time for the simulation is not too large. The step size must be much smaller than the dispersive and nonlinear lengths (see Section 5.1. Optical Solitons).

### 4.4. Order of the Simulation Program

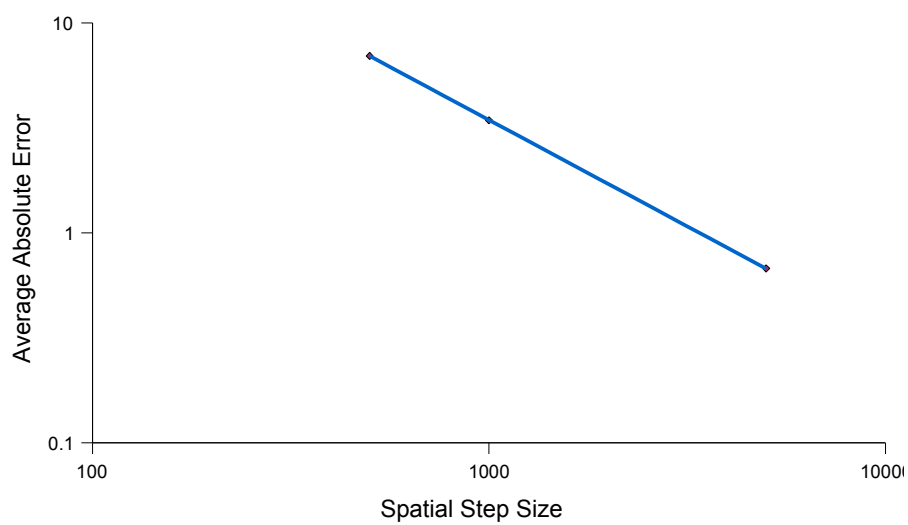
To find the actual order of the numerical method used in the program, a graph of the average absolute error was plotted against the number of spatial steps (nz) taken by the simulation.

$$\text{Average absolute error} = \frac{\sum_z^N |A(z) - A(0)|}{nz}$$

The order of a numerical method (n) is a measure of how the error changes with step size (dz).

$$\text{Error} \propto (dz)^n$$

$$n = \frac{\log(\text{Error})}{\log(dz)}$$



**Figure 5:** Average absolute error plotted against spatial step size.

Figure 5 shows that the order of the numerical simulation is one.

## 5. Testing the Numerical Simulation Program

The aim of this project is to create a numerical simulation model and use it to investigate how a supercontinuum would broaden towards the ultraviolet end of the spectrum. Before doing this the program had to be tested against existing experimental and theoretical results in the literature to ensure that the program was producing reliable results.

### 5.1. Optical Solitons

In the context of supercontinuum generation, an optical soliton is a packet of energy that propagates over long distances without becoming distorted. This occurs because the dispersive and nonlinear effects combine in such a way that the pulse does not change as it propagates, or its evolution is periodic along the fibre. They are also useful because they have analytical solutions (Agrawal, 2007).

The length of fibre over which dispersive and nonlinear effects are influential is defined by  $L_D$  and  $L_{NL}$  respectively;

$$L_D = \frac{T_0^2}{|\beta_2|} \quad L_{NL} = \frac{1}{\gamma P_0} \quad (17) \text{ \& \ } (18)$$

$T_0$  is the width of the pulse and is related to the full width half maximum (FWHM), which is the width of the pulse at half the maximum intensity of the pulse, by  $T_0 = 0.567 T_{FWHM}$ .  $\beta_2$  is the group velocity dispersion responsible for broadening the pulse.  $\gamma$  is the nonlinearity coefficient.

$P_0$  is the peak power of the input pulse.

Using Dudley et al. (2006), a third order soliton ( $N=3$ ) was simulated using the following initial pulse and fibre parameters:

$$T_0 = 28.4 \text{ fs}$$

$$P_0 = 1250 \text{ W}$$

$$\beta_2 = -1.183 \times 10^{-26} \text{ s}^2 \text{ m}^{-1}$$

$$\gamma = 0.11 \text{ W}^{-1} \text{ m}^{-1}$$

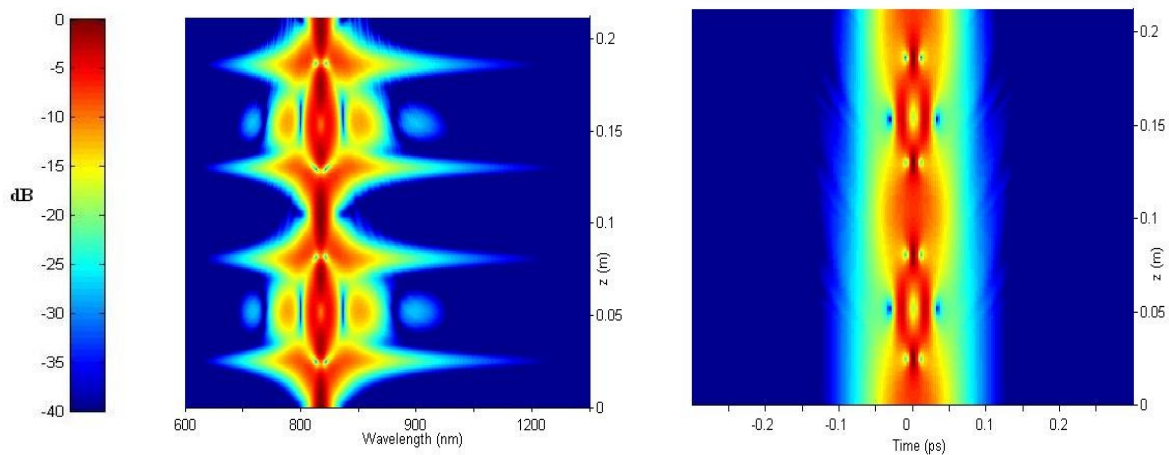
This gives  $L_D = 0.068 \text{ m}$ . Higher order dispersion was ignored and of the nonlinear effects only self-phase modulation (SPM) was taken into account ( $f_R = 0$  to exclude the other effects). For

solitons their period is equal to;

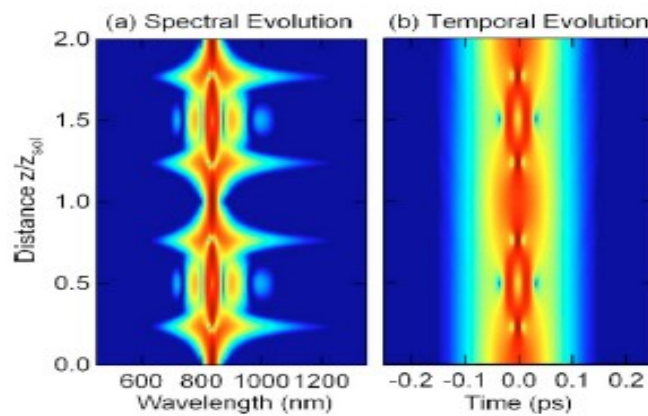
$$z_{sol} = \frac{\pi}{2} L_D \quad (19)$$

So one soliton period is equal, in this case, to 0.106 m. Figure 6 illustrates the results from the numerical simulations and Figure 7 is from Dudley et al. (2006). There is an excellent correlation between the simulation and the paper case. In both cases, and for the rest of this project, the intensity is shown on a decibel scale from 0 to -40. The equation for converting to decibels is shown in Equation (20). This means that -20 dB corresponds to an actual intensity of 1% of the maximum intensity. The x-axis for the figures is the spectral or temporal value and the y-axis is how far the pulse has propagated down the fibre.

$$intensity(dB) = 10 \log_{10} \left( \frac{intensity}{maximum\ intensity} \right) \quad (20)$$



**Figure 6:** Simulated spectral and temporal decibel plots of a 3<sup>rd</sup> order soliton ( $N=3$ ).

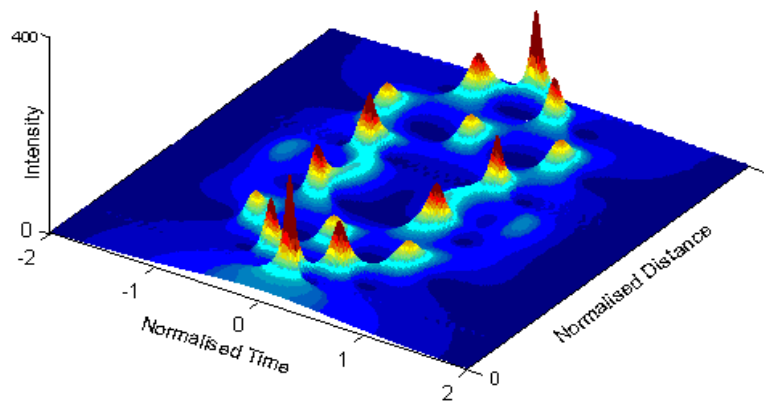


**Figure 7:** Spectral and temporal decibels plots of a 3<sup>rd</sup> order soliton (Dudley et al., 2006).

As a further example, Equation (21) shows the relation between soliton order and the dispersive and nonlinear lengths. This means that for the earlier case, if a fifth order soliton is to be simulated, then the power of the input pulse needs to be increased by 25/9 for the soliton to be stable.

$$N^2 = \frac{L_D}{L_{NL}} = \frac{\gamma P_0 T_0^2}{|\beta_2|} \quad (21)$$

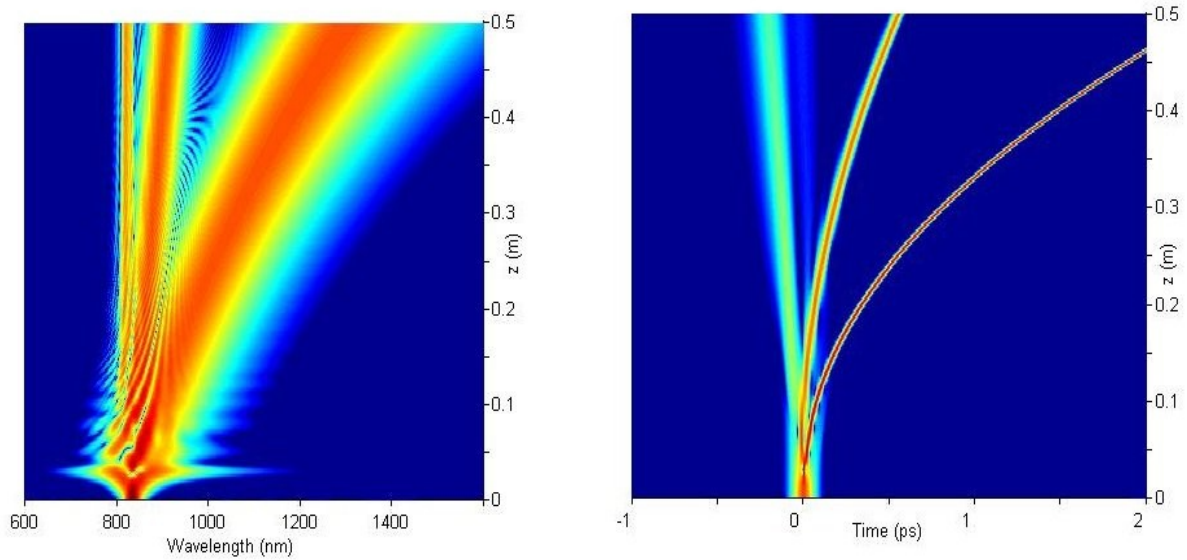
Figure 8 shows the temporal plot for a 5<sup>th</sup> order soliton, not on a decibel scale so that the intensity peaks can be more clearly seen.



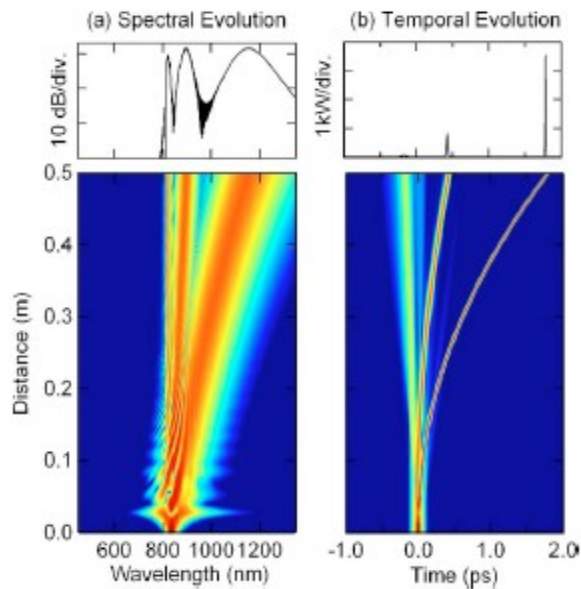
**Figure 8:** Temporal plot for a 5<sup>th</sup> order soliton.

## 5.2. Raman Scattering of Solitons

The next step in Dudley et al. (2006) was to include Raman scattering in the simulation to see its effect on solitons. The only change from the above case was to set  $f_R=0.18$  in the simulation so that Raman scattering would be included. Figure 9 shows the simulated spectral and temporal decibel plots of the Raman scattering that causes the soliton to break up. Figure 10 shows the results from Dudley et al. (2006).



**Figure 9:** *Simulated spectral and temporal plots of Raman induced fission of a soliton.*



**Figure 10:** *Spectral/temporal plots showing Raman induced soliton fission (Dudley et al., 2006).*

Figure 9 and Figure 10 match well. The fission is more exaggerated in the simulated case than for the paper result. This is most likely due to the sensitivity of the simulation to the Raman response function  $h_R(t)$ , as the paper used a different method to the one used in this project.

### 5.3. Comparison of Numerical Simulation with Paper Cases

In this example from Dudley et al. (2006), the supercontinuum is modelled using the full NLSE as described by Equation (5). The parameters used are as follows and the results are given by Figure

11, to be compared with the paper results shown in Figure 12.

*Initial Pulse Wavelength* = 835 nm

$\gamma = 0.11 \text{ W}^{-1} \text{ m}^{-1}$

*FWHM* = 50 fs

$P_0 = 10000 \text{ W}$

$\beta_2 = -11.83 \text{ ps}^2 \text{ m}^{-1}$

$\beta_3 = 8.1038\text{e-}2 \text{ ps}^3 \text{ m}^{-1}$

$\beta_4 = -9.5205\text{e-}5 \text{ ps}^4 \text{ m}^{-1}$

$\beta_5 = 2.0737\text{e-}7 \text{ ps}^5 \text{ m}^{-1}$

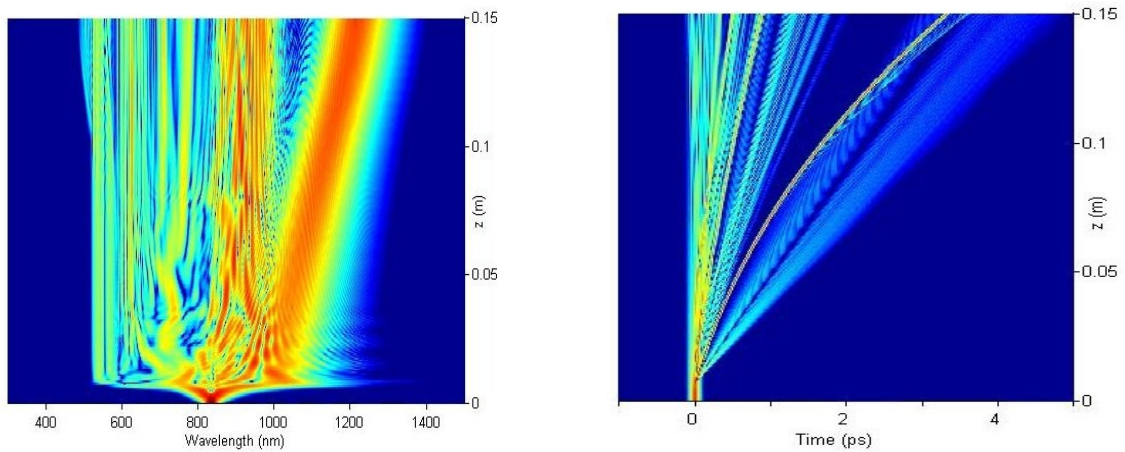
$\beta_6 = -5.3943\text{e-}10 \text{ ps}^6 \text{ m}^{-1}$

$\beta_7 = 1.3486\text{e-}12 \text{ ps}^7 \text{ m}^{-1}$

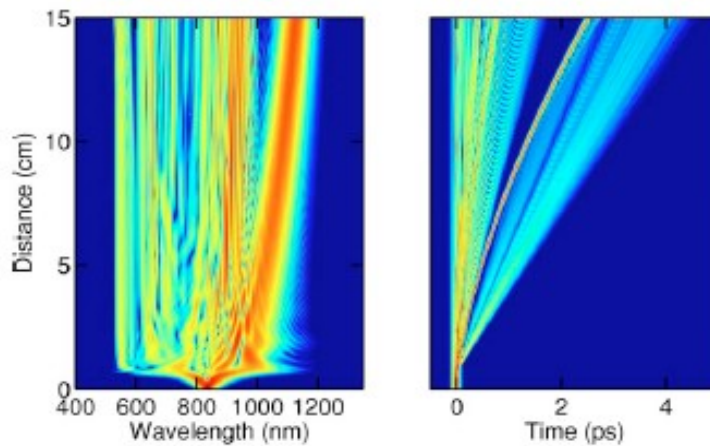
$\beta_8 = -2.5495\text{e-}15 \text{ ps}^8 \text{ m}^{-1}$

$\beta_9 = 3.0524\text{e-}18 \text{ ps}^9 \text{ m}^{-1}$

$\beta_{10} = -1.7140\text{e-}21 \text{ ps}^{10} \text{ m}^{-1}$



**Figure 11:** Simulated spectral and temporal plots.



**Figure 12:** Spectral and temporal plots of an example from Dudley et al. (2006).



## 6. Simulating Supercontinuum Generation in the Ultraviolet End of the Spectrum

### 6.1. Estimating Fibre Parameters

In order to generate a decent supercontinuum in the near-ultraviolet end of the electromagnetic spectrum (approximately 200 – 400 nm), the right optical fibre must be used. This means that the ZDW of the fibre must be in or around that region. No fibres with that low a ZDW currently exist so it is necessary for a hypothetical fibre with reasonably expected fibre parameters to be estimated. Data for a popular fibre with a ZDW of 750 nm were taken and extrapolated to a ZDW of 500 nm (this ZDW was chosen because the Laser Analytics Group has a laser that operates at a wavelength of 532 nm). This was done by taking the dispersion curve for the fibre and fitting a polynomial to that curve. The polynomial was then shifted from 750 nm to 500 nm. Dispersion is related to  $\beta_2$ , the GVD parameter, by the following equation from Agrawal (2007):

$$D = \frac{-2\pi c}{\lambda^2} \beta_2 \quad (22)$$

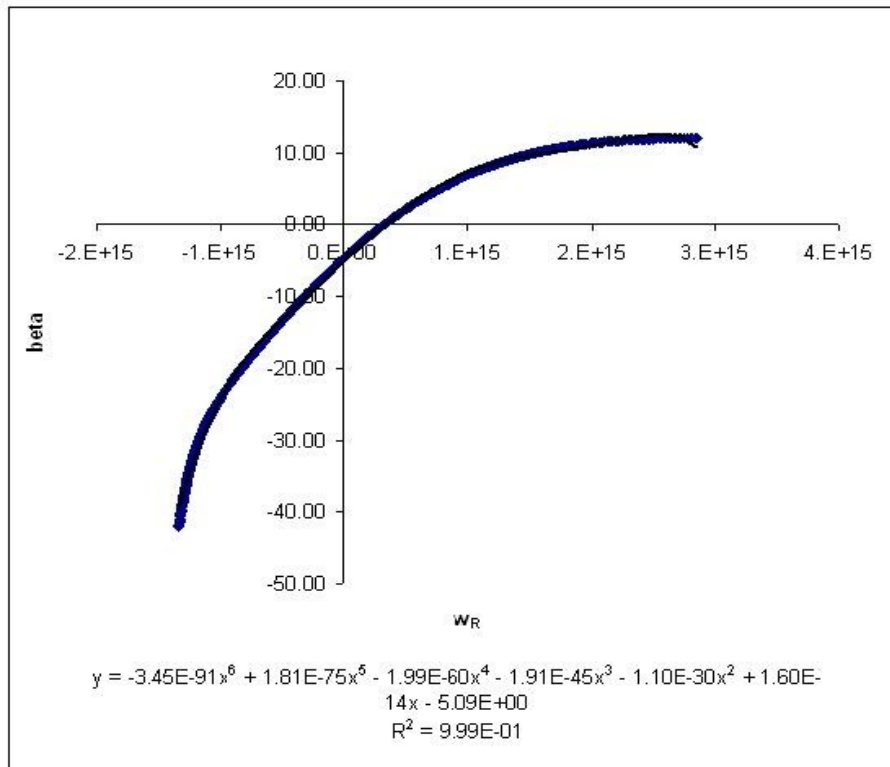
Using Equation (21), beta values could be calculated and plotted on a graph as done in Figure 13. A curve was then fitted to the data and the corresponding polynomial taken from it.

A Taylor expansion of beta to the 7<sup>th</sup> term is;

$$\beta_2(\omega) = \beta_2 + \beta_3 \omega_R + \beta_4 \frac{\omega_R^2}{2!} + \beta_5 \frac{\omega_R^3}{3!} + \beta_6 \frac{\omega_R^4}{4!} + \beta_7 \frac{\omega_R^5}{5!} + \beta_8 \frac{\omega_R^6}{6!} \quad (23)$$

where  $\omega_R = (\omega - \omega_0)$  and  $\omega_0$  is the frequency of the laser.

The polynomial in Figure 13 is matched to the Taylor expansion of Equation (23) to give the  $\beta$  values. The nonlinearity coefficient is held at  $\gamma = 0.11 \text{ W}^{-1} \text{ m}^{-1}$  as it does not tend to change much for these fibres (REF NL800 LIST).



**Figure 13:** *Beta curve for a 550 nm laser in a fibre with a 500 nm ZDW.*

## 6.2. Estimated Beta Values for Different ZDWs and Laser Wavelengths

Using the method outlined in section 6.1., beta values for certain laser pumping wavelengths at two different ZDWs (750 nm and 500 nm) were calculated. The results are shown in Table 1.

Pumping wavelengths 50 nm either side of each ZDW were used to test the effects on the supercontinuum of being either side of the ZDW. A pumping wavelength of 800 nm at a ZDW of 750 nm was used to test the sensitivity of the supercontinuum to changes in different variables, the results of which are discussed in the next section. The sensitivity analysis was done at this wavelength because fibres at this ZDW already exist, so the results can be checked easily. Finally, the pumping wavelength of 532 nm into a fibre with ZDW of 500 nm was simulated to see if supercontinua could be generated in the near-UV region.

**Table 1:** Beta values at different laser wavelengths and fibre ZDWs.

ZDW (nm)	500	500	500	750	750	750
Pumping Wavelength (nm)	450	532	550	700	750	800
beta 2	5.30E+00	-3.21E+00	-5.09E+00	1.19E+01	6.96E-01	-9.83E+00
beta 3	9.94E-03	1.58E-02	1.60E-02	5.90E-02	6.56E-02	6.78E-02
beta 4	-1.23E-05	-4.27E-06	-2.21E-06	-4.53E-05	-2.60E-05	-5.45E-07
beta 5	-3.27E-09	-1.84E-08	-1.15E-08	-8.24E-08	-1.35E-07	-1.90E-07
beta 6	4.54E-11	-1.90E-11	-4.78E-11	2.62E-10	3.23E-10	3.82E-10
beta 7	2.78E-14	2.08E-13	2.17E-13	-3.28E-13	-3.59E-13	-3.87E-13
beta 8	-2.48E-16	-2.99E-16	-2.48E-16	1.75E-16	1.75E-16	1.75E-16

### 6.3. Sensitivity Analysis of the Simulations to Changes in Variables

It is helpful to try and understand which variables could encourage supercontinuum growth. There are two sets that can be controlled; fibre parameters and laser source parameters.

**Table 2:** Variables that can be altered within the numerical simulation.

Fibre Parameters	Laser Source Parameters
Fibre Length, $z$ (m)	Wavelength, $\lambda$ (m)
Nonlinearity Coefficient, $\gamma$ ( $W^{-1} m^{-1}$ )	FWHM, (fs)
Dispersion Coefficients, $\beta_i$ ( $s^i m^{-1}$ )	Peak Intensity, $P_0$ (W)
	Shape of the Input Pulse

Generally speaking, the longer the fibre length, the broader the supercontinuum generated. Using the dispersion and nonlinearity lengths  $L_D$  and  $L_N$  allows a suitable length to be chosen. The fibre length  $z$  must be larger than  $L_D$  and  $L_N$  in order for the dispersive and nonlinear effects to broaden the inlet pulse.

The nonlinearity coefficient tends to be within a certain range for most current optical fibres, spanning 0.034 to 0.215  $W^{-1} m^{-1}$  (Blaze Photonics, 2007). Due to this uncertainty the nonlinearity coefficient was not changed from 0.11  $W^{-1} m^{-1}$ , and because many fibres have a coefficient in this region this is reasonable.

Dispersion Coefficients have already been estimated in Section 6.2..

The shape of the inlet pulse is secant in nature. Its form is given by Equation X and also in Appendix 1.3. This is the shape of the lasers used in the Laser Analytics Group.

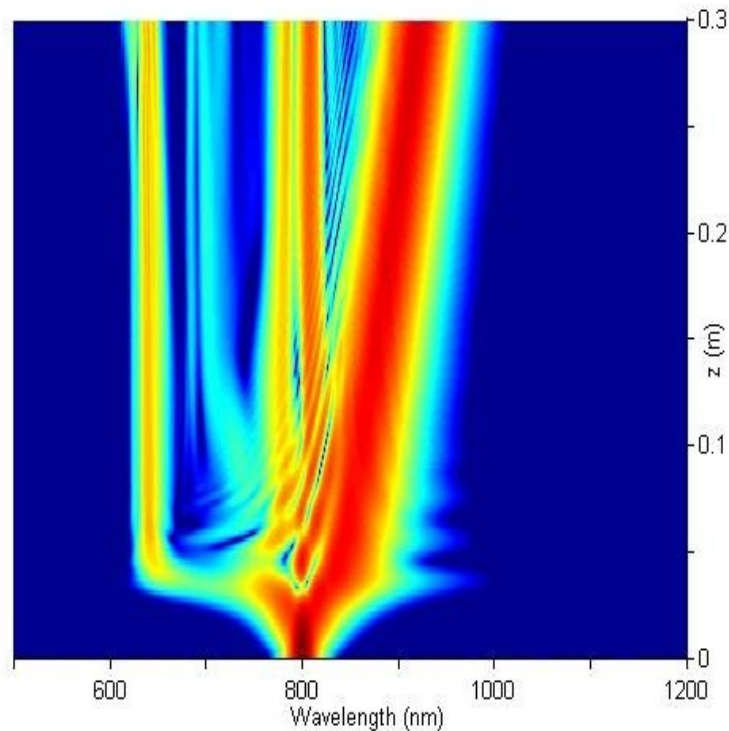
### 6.3.1. Base Case and Variable Changes for an 800 nm Laser

The parameters used for the base case (from which other variables will be changed one at a time to see the effect changing them has) are shown in Table 3.

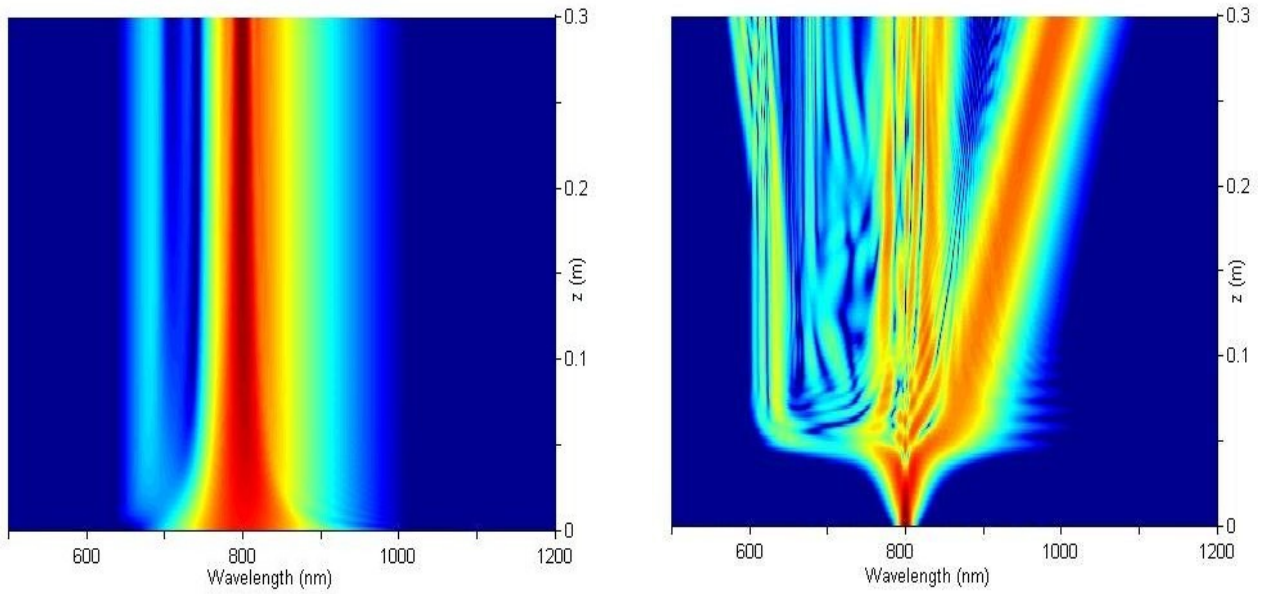
**Table 3:** Variables for the 800 nm laser base case.

Parameter	Value
Fibre Length, $z$ (m)	0.3
Nonlinearity Coefficient, $\gamma$ ( $\text{W}^{-1}\text{m}^{-1}$ )	0.11
Dispersion Coefficients, $\beta_i$ ( $\text{s}^i\text{m}^{-1}$ )	See Section 6.2
Wavelength, $\lambda$ (nm)	800
FWHM, (fs)	50
Peak Intensity, $P_0$ (W)	1000

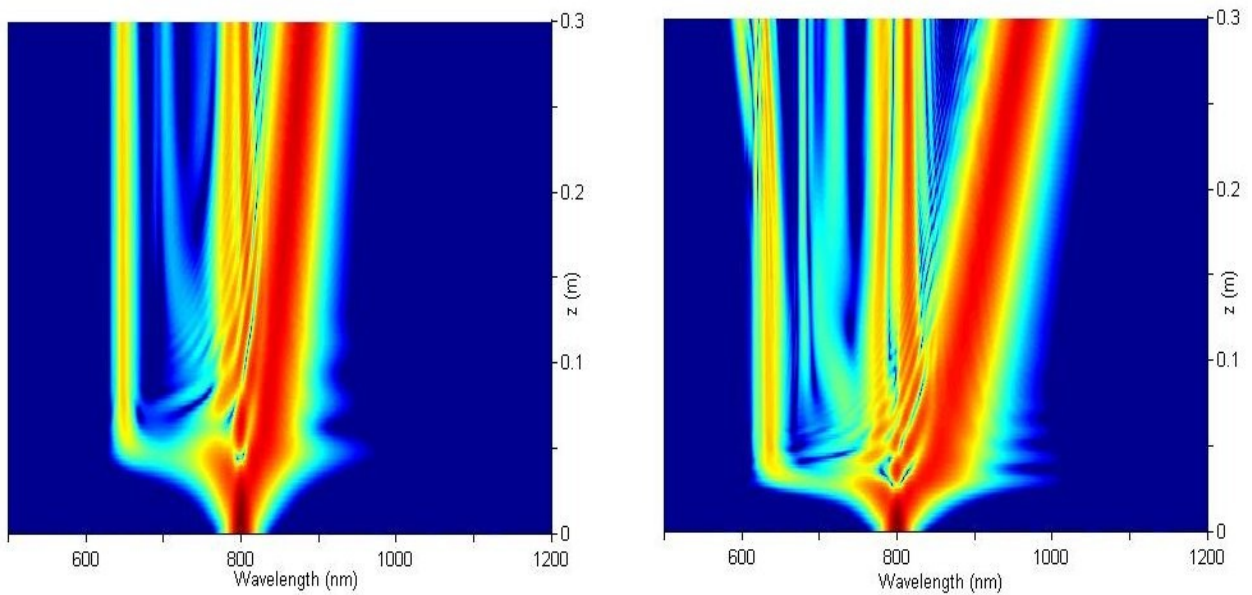
Figure 14 shows the supercontinuum generated by the base case. The following figures show what happens to the supercontinuum as individual parameters are changed. Note that when the laser pumping wavelength is changed, the dispersion coefficients must also be changed (see Section 6.2.).



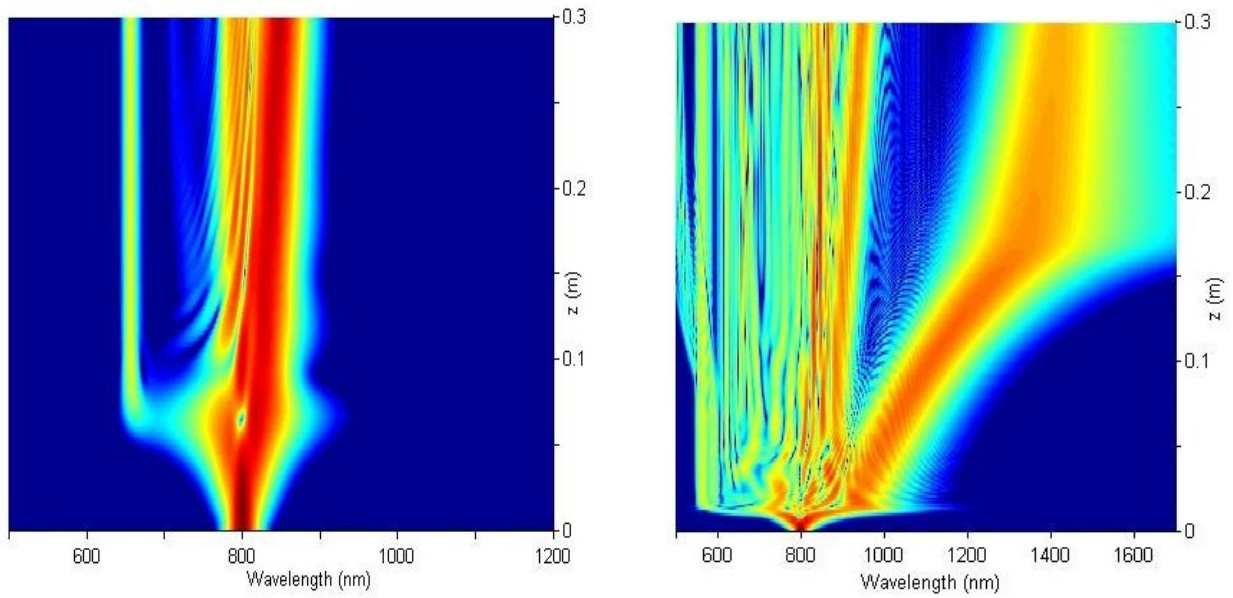
**Figure 14:** Base case for an 800 nm laser pumped into a fibre with a ZDW of 750 nm.



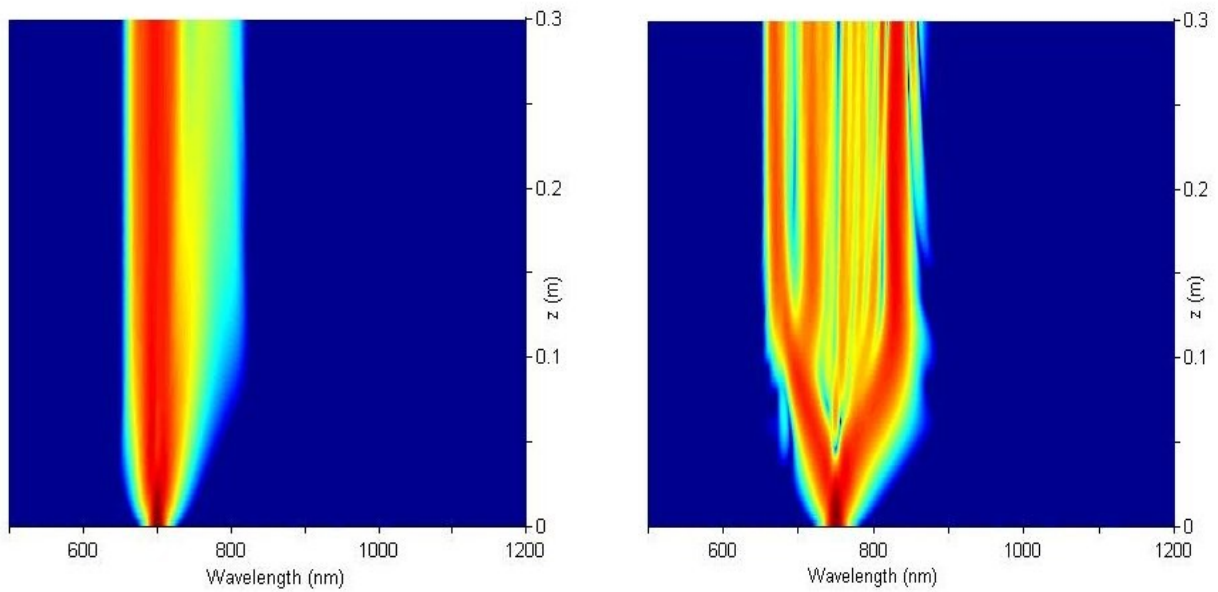
**Figure 15:** How changes in the FWHM affect the SC generated in a fibre with a ZDW of 750 nm. Left image has FWHM = 10 fs, right image has FWHM = 100 fs.



**Figure 16:** How changes in the nonlinearity coefficient affect the SC generated in a fibre with a ZDW of 750 nm. Left image has  $\gamma = 0.08 \text{ W}^{-1}\text{m}^{-1}$ , right image has  $\gamma = 0.14 \text{ W}^{-1}\text{m}^{-1}$ .



**Figure 17:** How changes in the power affect the SC generated in a fibre with a ZDW of 750 nm. Left image has power = 500 W, right image has power = 5000 W.



**Figure 18:** How changes in the pumping wavelength affect the SC generated in a fibre with a ZDW of 750 nm. Left image has wavelength = 700 nm, right image has wavelength = 750 nm.

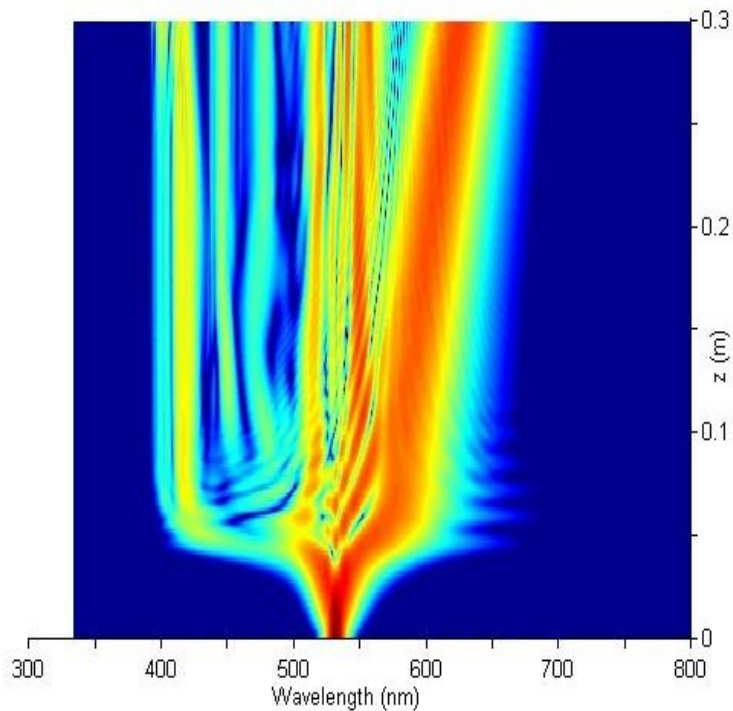
### 6.3.2. Base Case and Variable Changes for a 532 nm Laser

The parameters used for the base case (from which other variables will be changed one at a time to see the effect changing them has) are shown in Table 4.

**Table 4:** Variables for the 532 nm laser base case.

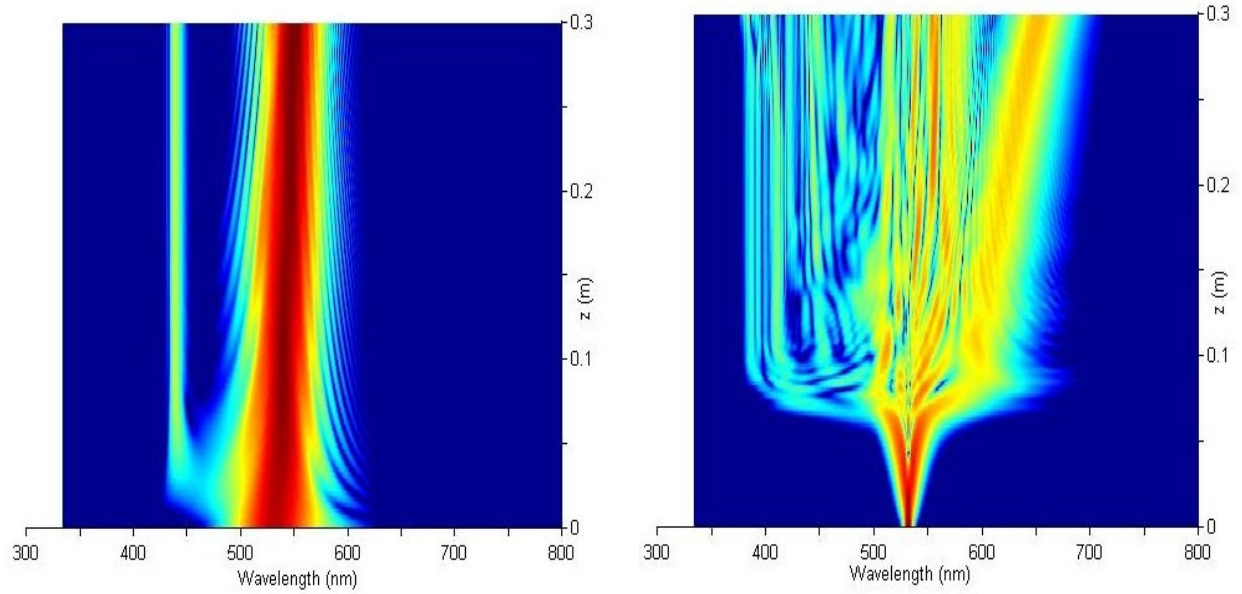
Parameter	Value
Fibre Length, $z$ (m)	0.3
Nonlinearity Coefficient, $\gamma$ ( $\text{W}^{-1}\text{m}^{-1}$ )	0.11
Dispersion Coefficients, $\beta_i$ ( $\text{s}^i\text{m}^{-1}$ )	See Section 6.2
Wavelength, $\lambda$ (nm)	532
FWHM, (fs)	50
Peak Intensity, $P_0$ (W)	1000

Figure 19 shows the supercontinuum generated by the base case. The following figures show what happens to the supercontinuum as individual parameters are changed.

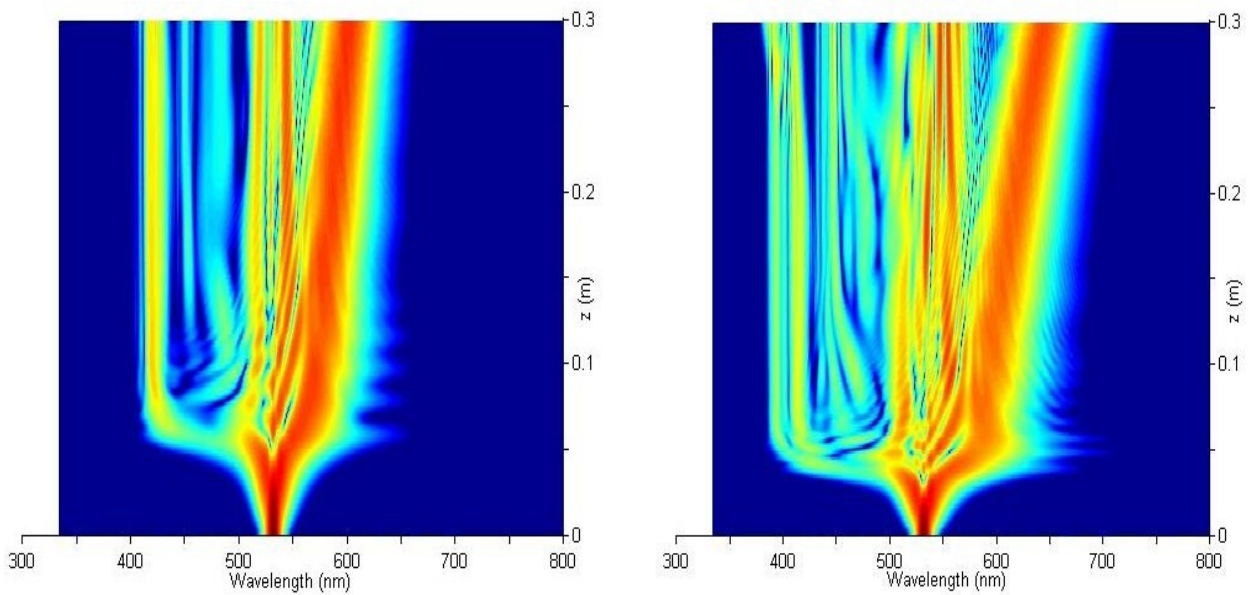


**Figure 19:** Base case for a 532 nm laser pumped into a fibre with a ZDW of 500 nm.



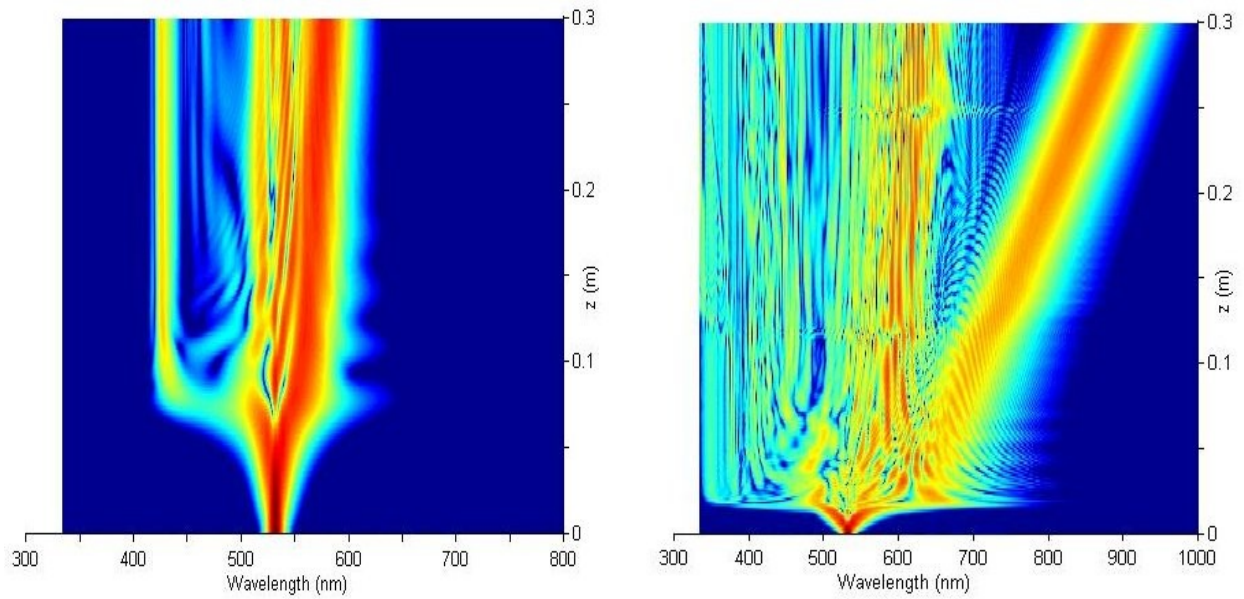


**Figure 20:** How changes in the FWHM affect the SC generated in a fibre with a ZDW of 500 nm. Left image has FWHM = 10 fs, right image has FWHM = 100 fs.

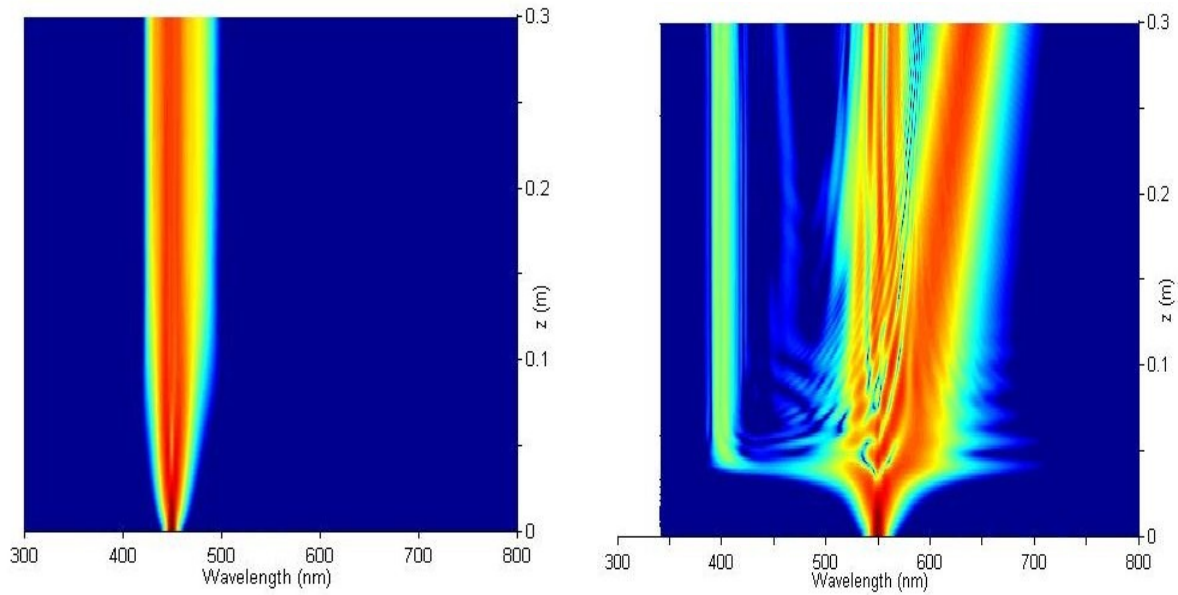


**Figure 21:** How changes in the nonlinearity coefficient affect the SC generated in a fibre with a ZDW of 500 nm. Left image has  $\gamma = 0.08 \text{ W}^{-1} \text{ m}^{-1}$ , right image has  $\gamma = 0.14 \text{ W}^{-1} \text{ m}^{-1}$ .





**Figure 22:** How changes in the power affect the SC generated in a fibre with a ZDW of 500 nm. Left image has power = 500 W, right image has power = 5000 W.



**Figure 23:** How changes in the pumping wavelength affect the SC generated in a fibre with a ZDW of 500 nm. Left image has wavelength = 450 nm, right image has wavelength = 550 nm.

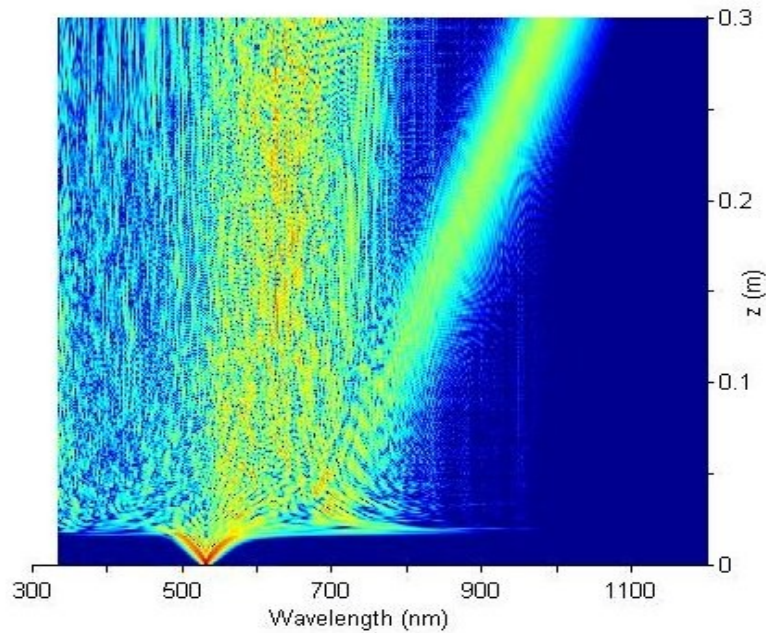
## 6.4. Simulating Broadening in the Ultraviolet Region of the Electromagnetic Spectrum

To maximise the broadening of the supercontinuum in the near-ultraviolet region of the spectrum, a high powered pulse going into a fibre with a ZDW of 500 nm was used. The parameters are shown in Table 5.

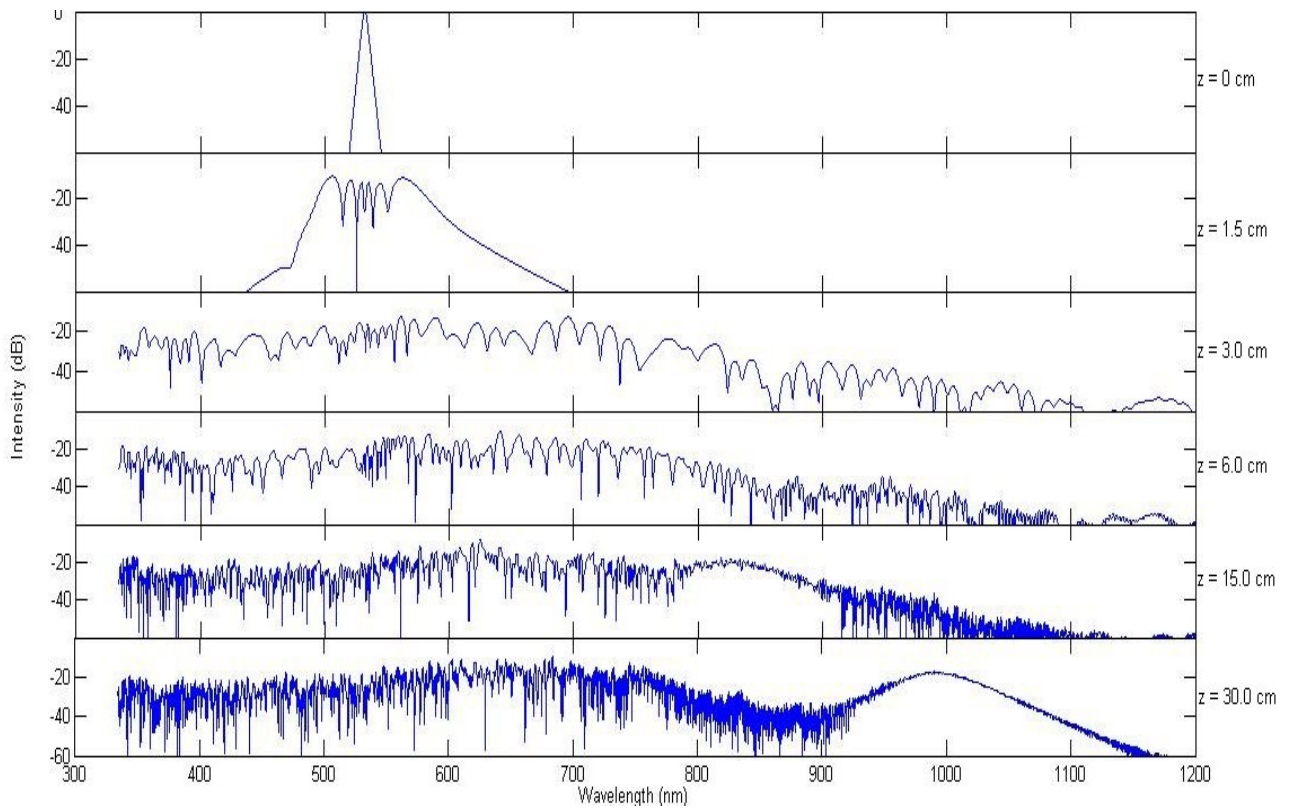
**Table 5:** Variables for the 800 nm laser base case.

Parameter	Value
Fibre Length, $z$ (m)	0.3
Nonlinearity Coefficient, $\gamma$ ( $\text{W}^{-1}\text{m}^{-1}$ )	0.11
Dispersion Coefficients, $\beta_i$ ( $\text{s}^i\text{m}^{-1}$ )	See Section 6.2
Wavelength, $\lambda$ (nm)	532
FWHM, (fs)	100
Peak Intensity, $P_0$ (W)	10000

Having the ZDW of the fibre just below the pumping wavelength means that dispersive effects are more significant. Figures 24 and 25 show the evolution of the pulse through the fibre.



**Figure 24:** Spectral broadening of a 532 nm laser propagating along a fibre with ZDW of 500 nm.



**Figure 25:** Evolution of the supercontinuum along a 0.3 m long fibre with a ZDW of 500 nm.

## **7. Discussion**

A higher FWHM, nonlinearity coefficient and pulse power all produce broader supercontinua. Pumping the laser at a wavelength around the ZDW of the fibre means that dispersive effects are small compared to nonlinear ones. At a wavelength lower than the ZDW of the fibre, the pulse does not experience much broadening. So it would be best to pump the laser into the fibre at a wavelength just above the ZDW of the fibre.

It can be seen from Figures 16 and 21 that higher nonlinearity coefficients cause greater break-up of the pulse. Higher pulse powers mean that the nonlinear effects are more emphasised (see the nonlinear length).

## **8. Proposals for Further Work**

The next step in this line of research would be mainly experimental. The results from the simulations need to be verified.

The numerical model can be improved. Some of the assumptions can be replaced by more rigorous theoretical or empirical findings. A better Raman response function could be employed, as used by Dudley et al. (2006). Instead of using a trapezoidal rule for the integration a higher order method such as a 4<sup>th</sup> order Runge-Kutta would be preferable. There is also no adaptive step size control in the program. Using one would enable less steps to be taken overall and would be more accurate.

## **9. Conclusions**

The project succeeded in generating a supercontinuum in the near-UV region of the spectrum. I gained a broad understanding of the nature of Optical Physics and became proficient in the use of Matlab by the end of the project.

The project also identified how changing variables affected the formation of the supercontinuum. It was found that using higher pulse power and FWHM, and fibres with a higher nonlinearity coefficient produced broader supercontinua.

## Appendix 1: *Matlab*<sup>TM</sup> Code

### Appendix 1.1: Main Code Used By A Typical Example

```
nt = 2^13;           % number of points
dt = 1.5e-15;       % timestep (dt) "0.01ps"
T = nt*dt;         % time window (period)
t = ((1:nt)'-(nt+1)/2)*dt; % time vector
w = wspace(T,nt);  % angular frequency vector
f = fftshift(w/(2*pi)); % frequency

z = 0.3;           % total distance
nz = 50000;       % total number of steps
nplot = 256;      % number of plots to make
n1 = round(nz/nplot); % number of steps per plot
nz = n1*nplot;    % total number of steps (revised)
dz = z/nz;        % step-size

lambda = 532e-9;   % wavelength of initial pulse (laser)
f0=3e8/lambda;
TR = 3e-15;

alpha = 0;
gamma = 0.11;

nanfirst = 0;
nansecond = 0;
nanthird = 0;

FWHM = 50e-15;
PeakIntensity = 1000;
Chirp = 0;

%beta(i) values in units --> s^(i) per metre.
beta0 = 0;
beta1 = 0;
beta2 = -3.21 *(1e-12)^2*1e-3;
beta3 = 1.58e-2 *(1e-12)^3*1e-3;
beta4 = -4.27e-6 *(1e-12)^4*1e-3;
beta5 = -1.84e-8 *(1e-12)^5*1e-3;
beta6 = -1.9e-11 *(1e-12)^6*1e-3;
beta7 = 2.08e-13 *(1e-12)^7*1e-3;
beta8 = -2.99e-16 *(1e-12)^8*1e-3;

betap = [beta0,beta1,beta2,beta3,beta4,beta5,beta6,beta7,beta8]; % dispersion polynomial
"using beta(i) from paper"

zv = (z/nplot)*(0:nplot);
```

```

u = zeros(length(t),length(zv));
ufourier = zeros(length(t),length(zv));

finderror = zeros(1, length(zv));
error = zeros(1, length(zv)-1);

% input pulse
u(:,1) = sechpulse(t,0,FWHM,PeakIntensity,Chirp);

sumsumfirstterm = 0;
sumsumsecondterm = 0;
sumsumthirdterm = 0;

tau1 = 12.2e-15;
tau2 = 32e-15;
hR = ((tau1^2+tau2^2)/(tau1*tau2^2)).*sin(t./tau1).*exp(-t./tau2);
Nkeep=round(300e-15/dt)+1;
hR=hR(nt/2:(nt/2+Nkeep));

figure(10)
plot(hR)

for ii = 1:nplot,
    [u(:,ii+1), nanfirst, nansecond, nanthird] =
    ssfm(u(:,ii),dt,dz,n1,alpha,betap,gamma,lambda,TR,hR,6,nanfirst,nansecond,nanthird, 1e-5);
    ufourier(:,ii) = fftshift(abs(dt*fft(u(:,ii))/sqrt(2*pi)).^2);
end

nanfirst
nansecond
nanthird

finderror = sum(u);

for ii = 1:(length(zv)-1),
    error(1,ii) = (finderror(1,ii) - finderror(1,(ii+1))) / finderror(1,ii);
end

En = sum((abs(u).^2));
Enn=En/En(1);

figure(1)
plot(Enn)

error = abs(error);

sprintf('sumsumfirstterm = %g, sumsumsecondterm = %g, sumsumthirdterm = %g',
sumsumfirstterm,sumsumsecondterm,sumsumthirdterm)

```

figure(2)

```
surf(zv,t,abs(u).^2,...  
     'MeshStyle','col','EdgeColor','none');  
set(gca,'YDir','reverse');  
xlabel('Z');  
ylabel('t');  
zlabel('abs(u).^2');
```

figure(3)

```
wavelength = (3e8)/(f+f0);
```

```
surf(zv,wavelength,ufourier,...  
     'MeshStyle','col','EdgeColor','none');  
set(gca,'YDir','reverse');  
hidden off;  
xlabel('Z');  
ylabel('lambda');  
zlabel('fft(abs(u).^2)');
```

figure(4)

```
surf(zv,wavelength,(10*log10(ufourier/(max(max(ufourier))))),...  
     'MeshStyle','col','EdgeColor','none');  
set(gca,'YDir','reverse');  
camup([1 0 0]);  
campos([z/2 lambda 1]);  
camtarget([z/2 lambda -20]);  
set(gca,'CLim',[-40 0]);  
set(gca,'XGrid','off');  
set(gca,'YGrid','off');  
set(gca,'TickDir','out');  
hidden off;  
xlim([0 z+0.0001]);  
xlabel('z (m)');  
set(get(gca,'XLabel'),'Rotation',90)  
ylabel('lambda (nm)');  
set(get(gca,'YLabel'),'Rotation',0.0);
```

figure(5)

```
plot(wavelength,(10*log10(ufourier(:,nplot)/(max(max(ufourier))))))
```

## Appendix 1.2: Code for the Function SSFM

```
function [u1, nanfirst, nansecond, nanthird] =  
ssfm(u0,dt,dz,nz,alpha,betap,gamma,wavelength,TR,hR,maxiter,nanfirst,nansecond,nanthird,tol);  
  
if (nargin<15)  
    tol = 1e-5;  
end  
  
w0 = (2*pi*3*10^8)/(wavelength);  
nt = length(u0);  
w = 2*pi*[0:nt/2-1,(-nt/2:-1)]/(dt*nt);  
fR = 0.18;  
  
Dhat = -alpha/2;  
for ii = 0:length(betap)-1;  
    Dhat = Dhat - j*betap(ii+1)*(w).^ii/factorial(ii);  
end  
Dhat = exp(Dhat*dz/2);  
  
u1 = u0;  
ufft = fft(u0);  
for iz = 1:nz,  
  
    uhalf = ifft(Dhat.*ufft);  
  
    for ii = 1:maxiter,  
        firstterm = ((abs(u1).^2 + abs(u0).^2);  
        secondterm = ((j./w0) .* (((1./u1).*gradient((abs(u1).^2).*u1).*dt) +  
((1./u0) .* gradient((abs(u0).^2).*u0).*dt)));  
  
        nan_region = find(isnan(secondterm));  
        nansecond = nansecond + sum(nan_region);  
        secondterm(nan_region) = 0;  
  
        nan_region = find(isnan(firstterm));  
        nanfirst = nanfirst + sum(nan_region);  
        firstterm(nan_region) = 0;  
  
        hR1 = hR;  
        hR1((length(hR)+1):nt) = 0;  
        thirdterm = ifft((fft(hR1).*fft(firstterm))*dt);  
  
        thirdterm = thirdterm + (j./(w0.*(u0))).*gradient(((u0).*thirdterm)).*dt +  
(j./(w0.*(u1))).*gradient(((u1).*thirdterm)).*dt;  
  
        nan_region = find(isnan(thirdterm));
```



```

nanthird = nanthird + sum(nan_region);
thirdterm(nan_region) = 0;

```

```

figure(100)
plot(abs(firstterm).^2)
hold on
plot(abs(secondterm).^2, 'r')
plot(abs(thirdterm).^2, 'g')

```

```

hold off

```

```

uv = exp(j*gamma*(((1-fR)*(firstterm + secondterm) + fR*thirdterm)*dz/2)) .* uhalf;

```

```

uv = fft(uv);
ufft = Dhat.*uv;
uv = ifft(ufft);
if (norm(uv-u1,2)/norm(u1,2) < tol)
    u1 = uv;
    break;
else
    u1 = uv;
end
end
if (ii == maxiter)
    warning(sprintf('Failed to converge to %f in %d iterations',...
        tol,maxiter));
end

```

```

u0 = u1;

```

```

end

```

### Appendix 1.3: Other Functions Used By The Main Code & SSFM

```

function w = wspace(t,nt);

```

```

dt = t/nt;
w = 2*pi*(0:nt-1)/t;
kv = find(w >= pi/dt);
w(kv) = w(kv) - 2*pi/dt;

```

```

function u = sechpulse(t,t0,FWHM,P0,C)

```

```

T0 = FWHM/(2*acosh(sqrt(2)));
u = sqrt(P0)*sech((t-t0)/T0).*exp(-i*C*(t-t0).^2/(2*T0^2));

```

## Appendix 2: Fourier Transforms

The issue of fourier transforms in *Matlab*<sup>TM</sup> and the scaling of them is described well by Usechak (2007).

The way the forward and reverse fourier transforms are defined within *Matlab*<sup>TM</sup> is different to the way they are defined by the supercontinuum community. The way in which this project uses fourier transforms is as follows, where FFT is the forward transform and IFFT is the reverse transform;

$$FFT[x(t)] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x(t) e^{i\omega t} dt \quad (24)$$

$$IFFT[X(\omega)] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} X(\omega) e^{-i\omega t} d\omega \quad (25)$$

However, *Matlab*<sup>TM</sup> defines its FFT and IFFT as follows;

$$FFT[x(t)] = \int_{-\infty}^{\infty} x(t) e^{-i\omega t} dt \quad (26)$$

$$IFFT[X(\omega)] = \int_{-\infty}^{\infty} X(\omega) e^{i\omega t} d\omega \quad (27)$$

This means that when the forward transform is used, the *Matlab*<sup>TM</sup> reverse transform must be used instead for the numerical simulation and scaled by  $\frac{dt}{\sqrt{2\pi}}$  where  $dt$  is the time window and scales the transformed matrix. This scale does not need to be used for data which is transformed and then reverse transformed.

### Appendix 3: Nomenclature

A	Complex Field Envelope	
c	Speed of Light	$\text{ms}^{-1}$
D	Dispersion	$\text{sm}^{-2}$
dz	Spatial Step Size	m
h	Spatial Step Size	m
$h_R(t)$	Raman Response Function	
$f_R$	Fraction of Nuclear Contribution to the Nonlinearity	
$L_D$	Dispersion Length	m
$L_{NL}$	Nonlinear Length	m
N	Soliton Order	
nZ	Number of Spatial Steps	
$P_0$	Inlet Peak Intensity	W
$R(t)$	Nonlinear Response Function	
t	Time	s
$t_e$	Time delay for the Electronic Contribution	s
$T_0$	Pulse Width	m
z	Length of Fibre	m
$\alpha$	Fibre Loss	$\text{Wm}^{-1}$
$\beta_i$	Dispersion Coefficients	$\text{s}^i\text{m}^{-1}$
$\gamma$	Nonlinearity Coefficient	$\text{W}^{-1}\text{m}^{-1}$
$\lambda$	Wavelength	m
$\tau_i$	Time Constants in the Raman Response Function	s
$\omega_0$	Input Pulse Carrier Frequency	Hz
$\omega$	Frequency Space or Frequency	Hz

## References

- Agrawal, G. P. (2007). *Nonlinear Fiber Optics*, Academic Press.
- Blaze Photonics (2007). NL-2.8-840. <http://www.crystal-fibre.com/datasheets/NL-28-840.pdf>.
- Blow, K.J. & Wood, D. (1989). Theoretical description of transient stimulated Raman scattering in optical fibers, *Quantum Electronics*, **25**, pp. 2665 -2673.
- Dudley, J. M. et al. (2006). Supercontinuum generation in photonic crystal fiber, *Reviews of Modern Physics*, **78**, Number 4.
- Fleck, J. A., Morris, J.R. & Feit, M.D. (1976). Time-dependent propagation of high energy laser beams through the atmosphere, *Appl. Phys. A*, **10**, pp 129-160.
- Hansen, K.P. & Kristiansen, R.E. (2007). Supercontinuum Generation in Photonic Crystal Fibres.
- Kibler, B. et al (2005). Supercontinuum generation and nonlinear pulse propagation in photonic crystal fiber: influence of the frequency-dependent effective mode area, *Appl. Phys. B*, **81**, p 337.
- Kobtsev, S.M. & Smirnov, S.V. (2006). Coherent properties of super-continuum containing clearly defined solitons, *Optics Express*, **14**, Number 9.
- Murphy, T.E. (2006). SSPROP - Split Step Fourier Propagation Software.  
<http://www.photonics.umd.edu/software/ssprop/>.
- Usechak, N. (2007). Figuring out Matlab's FFT/IFFT and their scaling.  
[http://xray.optics.rochester.edu:8080/users/noodles/papers/FFT\\_scaling.pdf](http://xray.optics.rochester.edu:8080/users/noodles/papers/FFT_scaling.pdf).
- Weiss, G.H. & Maradudin, A.A. (1962). The Baker-Hausdorff Formula and a Problem in Crystal Physics, *J. Math. Phys.*, **3**, pp 771-777.