Conventions for the Fourier Transform and NLS in OCS and Propagation Examples

John Zweck

March 10, 2003

1 The Fourier Transform

In FFTW the forward discrete Fourier transform $\mathbf{u} = \mathrm{FFT}_{\mathrm{FFTW}}(\tilde{\mathbf{u}})$ of a vector $\mathbf{u} = (u_0, \cdots, u_{N-1})$ is the vector $\tilde{\mathbf{u}} = (\tilde{u}_0, \cdots, \tilde{u}_{N-1})$, where

$$u_n \;=\; \sum_{n=0}^{N-1} \tilde{u}_m \exp(\frac{-2\pi i n m}{N}). \tag{1}$$

The inverse Fourier transform is $\mathrm{IFFT}_{\mathrm{FFTW}}$ where

$$\tilde{\mathbf{u}} = \frac{1}{N}\, \mathrm{IFFT}_{\mathrm{FFTW}}(\mathbf{u}). \tag{2}$$

In OCS we use a version of what Curtis calls the the physics convention. Let $u(t)$ be a function in the time domain that respresents the complex electric field envelope of the light. Suppose that $u(t)$ is periodic with period T. (In the code T = TimeWindow.) We define the Fourier transform of $u(t)$ to be $\tilde{u}(\omega)$, where ω is frequency in radians/second, by

$$\tilde{u}(\omega) \;=\; \frac{1}{T}\int_0^T u(t)\exp(i\omega t)dt. \tag{3}$$

So, by the Fourier inversion theorem,

$$u(t) \;=\; \frac{T}{2\pi}\int_0^\Omega \tilde{u}(\omega)\exp(-i\omega t)dt, \tag{4}$$

where $\frac{\Omega}{2\pi}$ = FrequencyWindow in Hz. The reason we use these normalizations is that then $|u(t)|^2$ and $|u(\omega)|^2$ both have units of Watts, W.

We now discretize $u(t)$ giving a length N vector $\mathbf{u}$ and $\tilde{u}(\omega)$ giving a vector $\tilde{\mathbf{u}}$. So if $\frac{\Omega}{2\pi}$ = FrequencyWindow (in OCS) then $T\frac{\Omega}{2\pi} = N$ and $\Delta\nu = \frac{\Delta\omega}{2\pi}$ = DeltaFreq (in OCS) which has units of Hz.

With this discretization

$$\mathbf{u} = \mathrm{FFT}_{\mathrm{FFTW}}(\tilde{\mathbf{u}}) = \mathrm{IFFT}_{\mathrm{OCS}}(\tilde{\mathbf{u}}) \tag{5}$$

where $\mathrm{IFFT}_{\mathrm{OCS}}$ is the method cfftw::IFFT in the OCS class cfftw. Similarly,

$$\tilde{\mathbf{u}} = \frac{1}{N}\,\mathrm{IFFT}_{\mathrm{FFTW}}(\mathbf{u}) = \mathrm{FFT}_{\mathrm{OCS}}(\mathbf{u}) \tag{6}$$

where $\mathrm{FFT}_{\mathrm{OCS}}$ is the method cfftw::FFT in the OCS class cfftw. With these definitions

$$\mathrm{FFT}_{\mathrm{OCS}} \circ \mathrm{IFFT}_{\mathrm{OCS}} = \mathrm{Id} \tag{7}$$

is the identity operator.

2 Energy, Average Power and Power Spectral Density

We have the following definitions and facts which are used in the OptSignal class in OCS.

$$\text{Total Energy (in J = W.s)} \;=\; \int_0^T |u(t)|^2 dt \;\approx\; \sum_n |u_n|^2 \Delta t \tag{8}$$

and

$$\text{Average Power (in W = J/s)} \;=\; \frac{1}{T}\int_0^T |u(t)|^2 dt \;\approx\; \frac{1}{N}\sum_n |u_n|^2 \tag{9}$$

so that Total Energy = T Average Power. Given our definitions, Parsevals Theorem says that the total energy is also

$$\int_0^T |u(t)|^2 dt \;=\; \frac{T^2}{2\pi}\int_0^\Omega |\tilde{u}(\omega)|^2 d\omega \tag{10}$$

so

$$\text{Total Energy} \;=\; T\sum_m |\tilde{u}_m|^2. \tag{11}$$

Also by Parseval,

$$\text{Average Power} \;=\; \sum_m |\tilde{u}_m|^2. \tag{12}$$

We define power spectral density to be avergae power in a bandwidth divided by the bandwidth in Hz. So

$$\mathrm{PSD}_m \;=\; \frac{1}{\Delta\nu^2}\int_{2\pi m\Delta\nu}^{2\pi(m+1)\Delta\nu} |\tilde{u}\omega)|^2 \frac{d\omega}{2\pi} \;\approx\; \frac{|\tilde{u}_m|^2}{\Delta\nu} \tag{13}$$

in W/Hz. Then it is easy to check that

$$\text{Average Power} \;=\; \sum \mathrm{PSD}_m \Delta\nu. \tag{14}$$

3 NLS and the Symmetric Split-step Scheme

With the physics convention the NLS is

$$iu_z \; - \; \tfrac{1}{2}\beta'' u_{tt} \; - \; \tfrac{i}{6}\beta''' u_{ttt} \; + \; \gamma|u|^2 u - i\alpha u = 0. \tag{15}$$

In the frequency domain the dispersion is done using a Taylor's series expansion and so

$$\tilde{u}(z+\Delta z,\omega) \; = \; \tilde{u}(z,\omega)\exp(\tfrac{i}{2}\beta''\omega^2\Delta z + \tfrac{i}{6}\beta'''\omega^3\Delta z). \tag{16}$$

Nonlinearity in the time domain is done as

$$u(z+\Delta z,t) \; = \; u(z,t)\exp(i\gamma|u|^2\Delta z). \tag{17}$$

In OCS we use the symmetric split-step Fourier scheme. This means that for a step of size Δz is performed as follows:

$$\begin{aligned}
\tilde{u}_{D1}(\omega) &= \tilde{u}(z,\omega)\exp([\tfrac{i}{2}\beta''\omega^2 + \tfrac{i}{6}\beta'''\omega^3]\tfrac{\Delta z}{2}) && (18)\\
u_{D1}(t) &= \mathrm{IFFT}_{\mathrm{OCS}}(\tilde{u}_{D1}(\omega)) && (19)\\
u_{\mathrm{NL}}(t) &= u_{D1}(t)\exp(i\gamma|u_{D1}(t)|^2\Delta z) && (20)\\
\tilde{u}_{\mathrm{NL}}(\omega) &= \mathrm{FFT}_{\mathrm{OCS}}(u_{\mathrm{NL}}(t)) && (21)\\
\tilde{u}(z+\Delta z,\omega) &= \tilde{u}_{\mathrm{NL}}(\omega)\exp([\tfrac{i}{2}\beta''\omega^2 + \tfrac{i}{6}\beta'''\omega^3]\tfrac{\Delta z}{2}) && (22)\\
u(z+\Delta z,t) &= \mathrm{IFFT}_{\mathrm{OCS}}(\tilde{u}(z+\Delta z,\omega)). && (23)
\end{aligned}$$

With this scheme, the local error is $O(\Delta z^3)$ and the global error is $O(\Delta z^2)$. For more details on errors see the paper of Oleg Sinkin, Ron Holzlöhner *et al.* on the course web page, SplitStep.pdf.gz

In the OCS classes `OptFiberLocalError` and `ocsOptFiber` instead of specifying β'' in $\mathrm{ps}^2/\mathrm{km}$ and β''' in $\mathrm{ps}^3/\mathrm{km}$ we specify `DispersionFiber` in ps/(nm.km) and `DispSlope-Fiber` in ps/(nm^2.km). Of course the value of `DispersionFiber` depends on a reference wavelength. We can either specify `ReferenceFreq` or `ReferenceWavelength` by setting the other value to 0.0. The formulae used to get the β's are given in the method `OptFiber-LocalError::SetDispersion`. (Also see Agrawal's Optical Fiber Communications book.) Note that $\beta'' =$ `FstOrDispFiber` and $\beta''' =$ `SndOrDispFiber`.

The nonlinear coefficient γ is computed from the input parameters `NonLinIndexFiber` in m^2/W and `EffectAreaFiber` in m^2 using the formula

$$\gamma = \frac{\mathtt{NonLinIndexFiber} * 2 * \pi * \mathtt{CenterFreq}}{\mathtt{EffectAreaFiber} * \mathtt{LightSpeed}}. \tag{24}$$

4 Step Size Selection Criteria in the class `OptFiberLocalError`

For more details on step size selection criteria, see the paper of Oleg Sinkin, Ron Holzlöhner *et al.* on the course web page, SplitStep.pdf.gz.

In `OptFiberLocalError` we have four step size selection options. The input parameter `TypeStepSizes` can take the values LOCAL_ERROR_3RD = 1, LOCAL_ERROR_2ND = 2, WALK_OFF = 3, CONSTANT = 4.

No matter what method we use we always choose the last step in a particular fiber so that the the total propagation length in that fiber is correct.

The rest of this section applies whenever `TypeSolver` is either SCALAR_NLS or VECTOR_MANAKOV_NO_PMD. The option VECTOR_MANAKOV_NO_PMD should be used when the PMD in the system is negligible, but you want to use a simulation with two orthogonal polarization states to model the optical noise correctly when doing Monte Carlo simulations. The third option for `TypeSolver` is VECTOR_MANAKOV_PMD, which has not been carefully debugged yet. (For PMD modeling, use the class `OptFiber` instead.)

4.1 CONSTANT method

In this method the step sizes are constant. The number of steps per fiber is `NumStepsBetweenScatterings`. (This variable name was chosen for the case when there is PMD. Hence the funny name.)

4.2 WALK_OFF method

The code uses a variant of the criterion for choosing Δz given in Oleg's split step paper. We input the `WalkOffParameter` and choose

$$\Delta z = \frac{\texttt{WalkOffParameter}}{|D_1\lambda_1 - D_2\lambda_2|} \tag{25}$$

where D_1 and D_2 are the dispersions in ps/nm-km at wavelengths λ_1 and λ_2. We choose λ_1, λ_2 as follows:

- WDM Case: λ_1, λ_2 are the central wavelengths of the edge channels
- Single Channel Case: $\lambda_{1,2}$ correspond to frequencies $\omega_{1,2} = \texttt{CenterFreq} \pm$ RootMeanSquare spectral width of signal.

4.3 LOCAL_ERROR methods

See Oleg's paper for the basic algorithm. Here we just talk about issues that are not mentioned in the paper.

For this method you need to set `RelativeErrorGoal`, and `DeltaZMax`. Note that `MaxPhaseChangeDeg` is an old parameter that is not used any more.

The algorithm aims to keep the local error in the range $0.5 * \texttt{RelativeErrorGoal} < \texttt{LocalError} < 2 * \texttt{RelativeErrorGoal}$. To determine the initial step size in a given fiber for the very first time through that fiber we take an initial guess of `DeltaZ` = 0.5*`DeltaZMax`. Note though that the algorithm may not actually take a step of that size. For subsequent passes through the fiber we use the step size of the first step that was taken in the fiber last time we went through it.

For LOCAL_ERROR_3RD for each step we keep the high-order solution given by Eq. (9) of Oleg's split step paper. For LOCAL_ERROR_2ND we keep the fine step solution, given by Eq. (8) of Oleg's split step paper.

5 Tests

5.1 Soliton Propagation

If we propagate a single sech pulse $u(t)$ the power function $|u(t)|^2$ should be constant in z but the phase will evolve in z. Note that classical solitons only exist when $\beta'' < 0$. We use the equation

$$iu_z \; - \; \tfrac{1}{2}\beta'' u_{tt} \; + \; \gamma |u|^2 u = 0 \tag{26}$$

with $\beta'' = -0.1\ \mathrm{ps}^2/\mathrm{km}$, $\gamma = 2.2\ \mathrm{W}^{-1}\ \mathrm{km}^{-1}$ and an initial pulse

$$u(z=0,t) \; = \; \sqrt{\frac{-\beta''}{\gamma\tau^2}}\,\mathrm{sech}\left(\frac{t}{\tau}\right) \tag{27}$$

where $\mathrm{FWHM} = 2\log(1+\sqrt{2})\tau = 4.0$ ps. The peak power is $P_0 = 8.8273789$ mW. The dispersive and nonlinear lengths are both equal to 51.493 km. The formula for the solution is

$$u(z,t) \; = \; \sqrt{\frac{-\beta''}{\gamma\tau^2}}\,\mathrm{sech}\left(\frac{t}{\tau}\right)\,\exp\left(i\frac{-\beta'' z}{2\tau^2}\right). \tag{28}$$

In Fig. 1, the optical power $|u(t)|^2$ of the soliton is shown at 0 km and 500 km.

In Fig. 2 we propagate the initial pulse in Eq. (27) for 300 km including only dispersion in the propagation model. The split-step method uses just 1 step since there is no nonlinearity.

In Fig. 3 we show the same pulse after 500 km of propagation through fiber which only has dispersion. The result is not correct since the pulse is wider than the `TimeWindow` and so has wrapped around on itself. To overcome this problem you would need to make `StringLength` and `qtPoints` both larger.

In Fig. 4, the real and imaginary parts of u are shown at 10,000 km. The agreement between theory and simulation is excellent. For this simulation we used `qtPoints` = 4096, `StringLength` = 4, `BitRateChann` = 40 GHz, and constant Step sizes with 50,000 steps. It is very important that the peak power in the simulation and theory agree as accurately as possible. Likewise for the FWHM.

5.2 Pure Third Order Dispersion

If we have no nonlinearity, $\beta'' = 0$ and $\beta''' = 1\ \mathrm{ps}^3/\mathrm{km}$ for 100 km and an initial Gaussian pulse with FWHM = 4 ps then the solution is shown in Fig. 5. The oscillations are in positive time as we would expect.

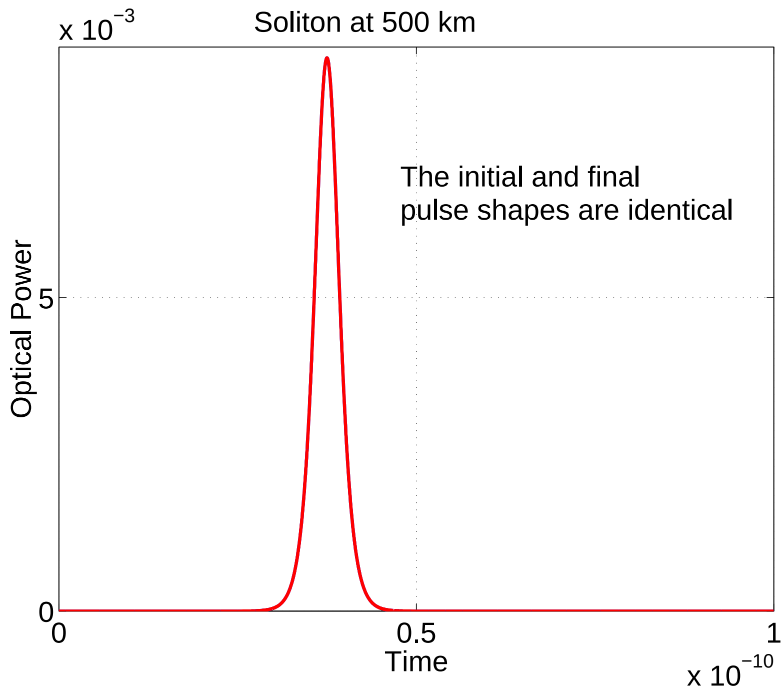

Figure 1: Soliton at 0 km and 500 km

5.3 Frequency shift invariance

We simulate the propagation of a Gaussian pulse at $\nu_{\rm ch} = 193400$ GHz through 50 km of fiber with $D = 20$ ps/nm-km. There is no dispersion slope or nonlinearity. We choose `qtPoints` = 4096 and `StringLength` = 32 to give `FrequencyWindow` = 1280 GHz.

We do two simulations. For the first we choose `CenterFreq` = 193400 GHz and for the second `CenterFreq` = 193300 GHz. The outputs are shown in Fig 6. Both have the same shape but the second is moved by $\Delta t = -801.5$ ps. This time shift is given by the formula

$$\Delta t = 2\pi\beta'' z\Delta\nu, \tag{29}$$

where $\Delta\nu = \nu_{\rm ch} - \nu_0 = +100$ GHz and $\beta'' = -25.513$ ps^2/km. In particular, the signs of the time shifts are the same in the theory and with the NLS simulation. One way to derive the theoretical time shift is to use the solution of the linear dispersive equation with a Gaussian initial condition as one has in the Physics convention

$$u(z=0,t) \;=\; \exp(-t_*^2/2)\exp(-i\Delta\nu_* t_*) \tag{30}$$

where $t_* = t/\tau_0$ and $\Delta\nu_* = \Delta\nu\tau_0$ are unitless and τ_0 is a measure of the pulsewidth. The solution of the linear dispersive equation

$$iu_z \;-\; \tfrac{1}{2}\beta'' u_{tt} = 0 \tag{31}$$

is

$$u(z,t) \;=\; \frac{1}{\sqrt{1-iB_*(z)}}\exp\left(-\frac{(t_*+i\Delta\nu_*)^2}{2(1-iB_*(z))}\right)\exp(-\Delta\nu_*^2/2), \tag{32}$$

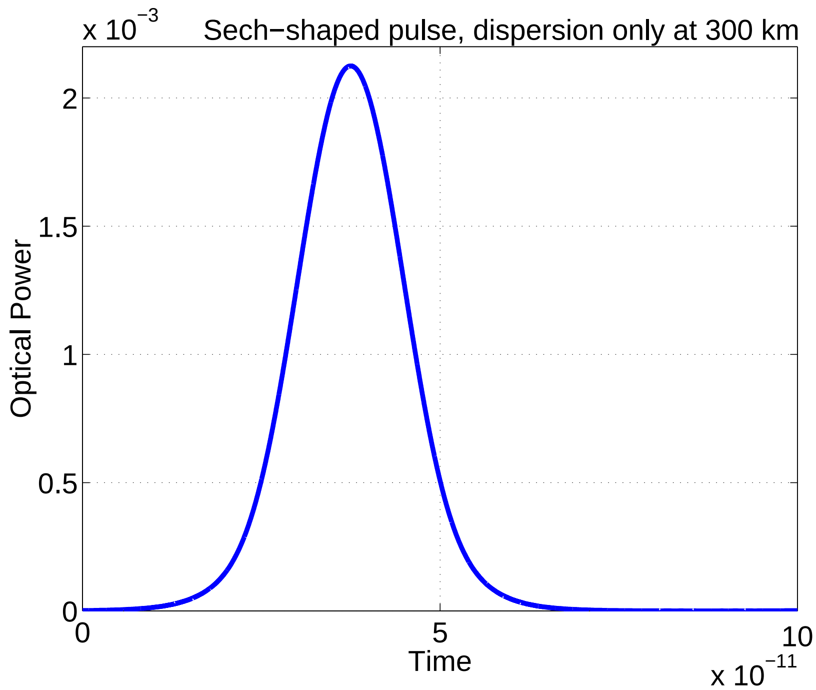

Figure 2:

where $B_*(z) = \int \beta''(w)dw/\tau_0^2$ is the normalized accumulated dispersion. From Eq. (32) it is easy to verify Eq. (29)

5.4 Tyco System

In Fig. 7 we plot the optical power versus time for propagation of a noise-free 32 bit PRBS through the Tyco system. The input pulses have a peak power of 1 mW. The corresponding electrical eye is shown in Fig. 9. Plots of optical power for an isolated pulse versus time at the end of each span of the dispersion map are shown in Fig. 8. The initial pulse width ($z = 0$) is about 30 ps, and after the pre-compensation it is about 200 ps. The final pulse width is about 22 ps.

I compared the Tyco system with $\beta''' = 0$ in the Math and Physics conventions before the electrical filter. The results are identical (I diff'ed the files) as one would expect. After consulting with Ron, I changed a sign in ElecFilter::Bessel5 and ElecFilter::Bessel4, to ensure that, apart from Ron's time-centering time shift, the filters are causal. The eye diagram for the electrical filtered Tyco system is shown in Fig. 9. Notice that the left side of the eye is steeper than the right side, as one would expect for a causal filter. (Causality implies that if the signal is zero for all times less than t_0 then the filtered signal must be zero there too.)

I also checked that the clock recovery time, Q-factor and Eye Opening look correct, even when the channel in question is not the central channel. In fact I checked that the Q-factor and Eye Opening are invariant to shifts in the central frequency of the FreqWindow.

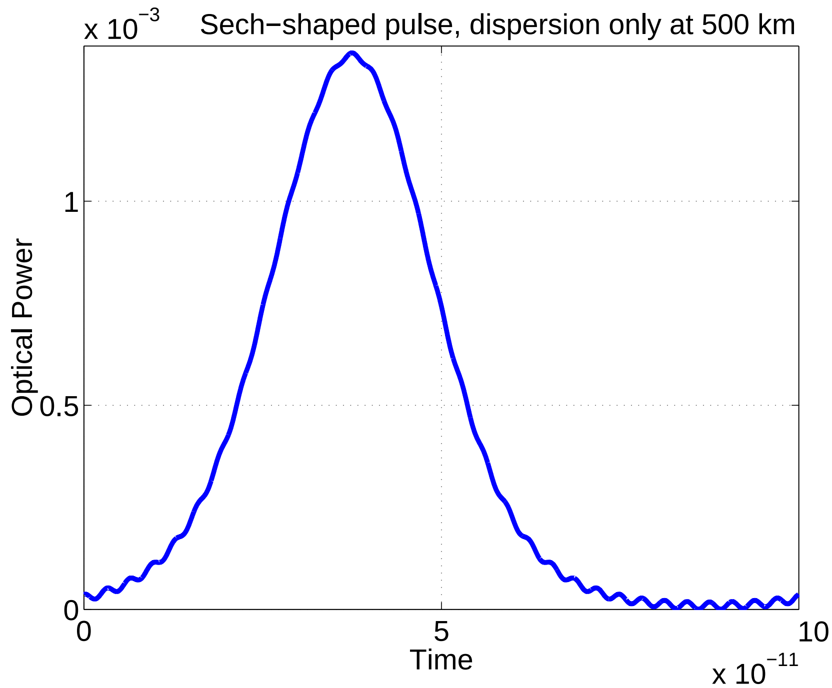

Figure 3:

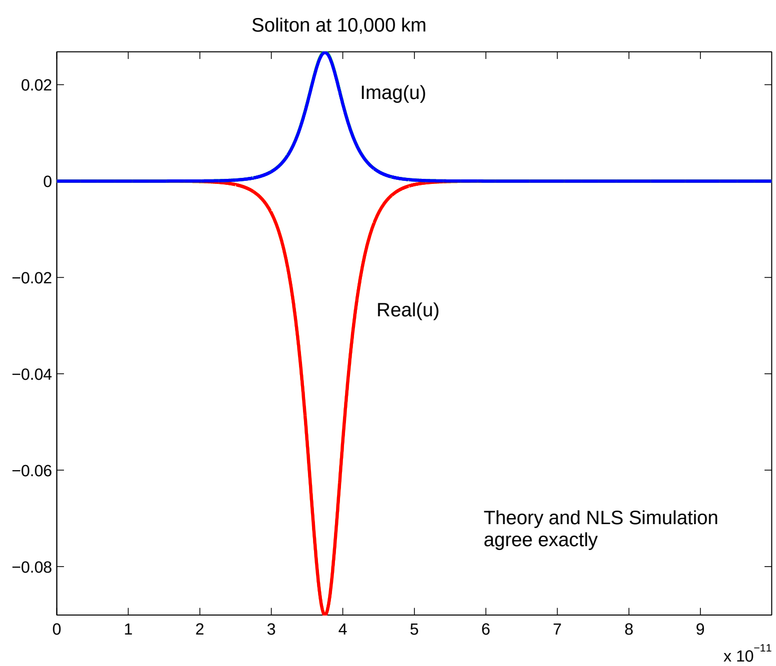

Figure 4: Soliton

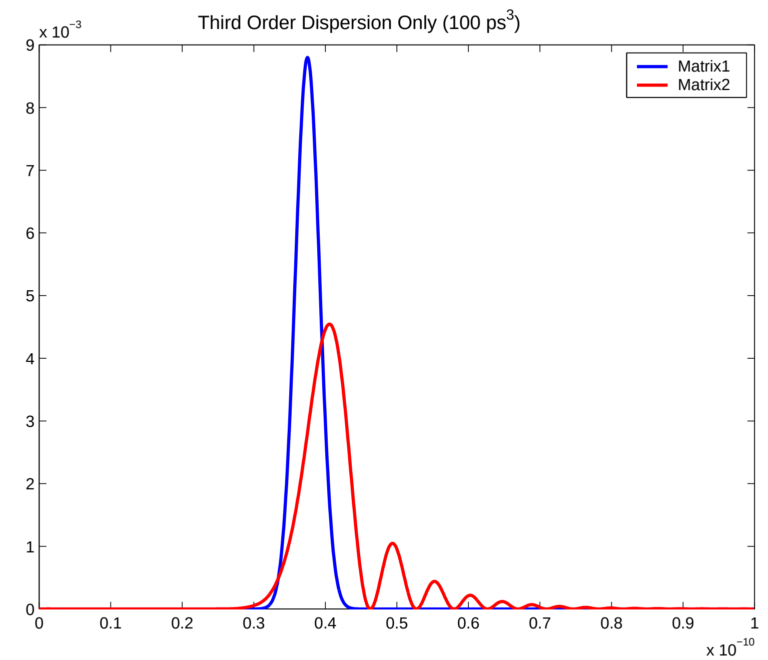

Figure 5: Third order dispersion

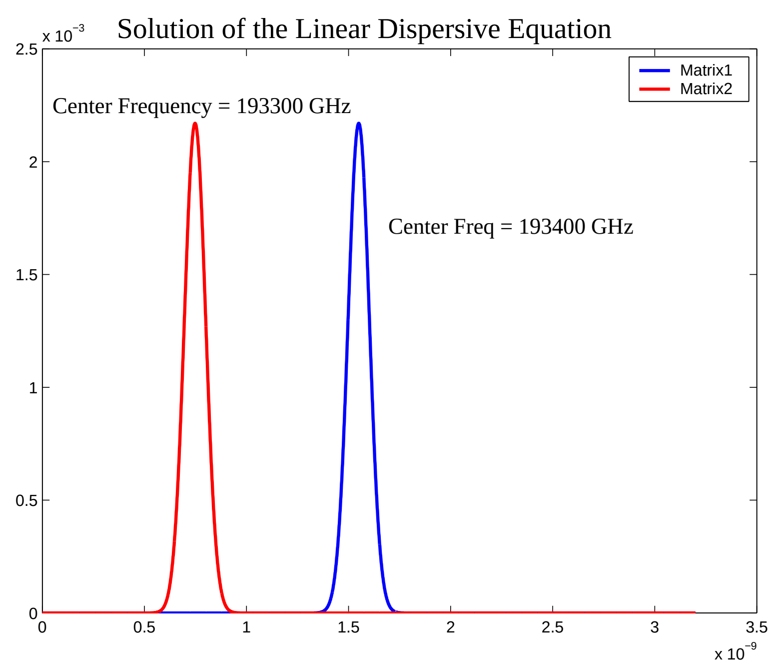

Figure 6: Frequency-shift invariance of the NLS

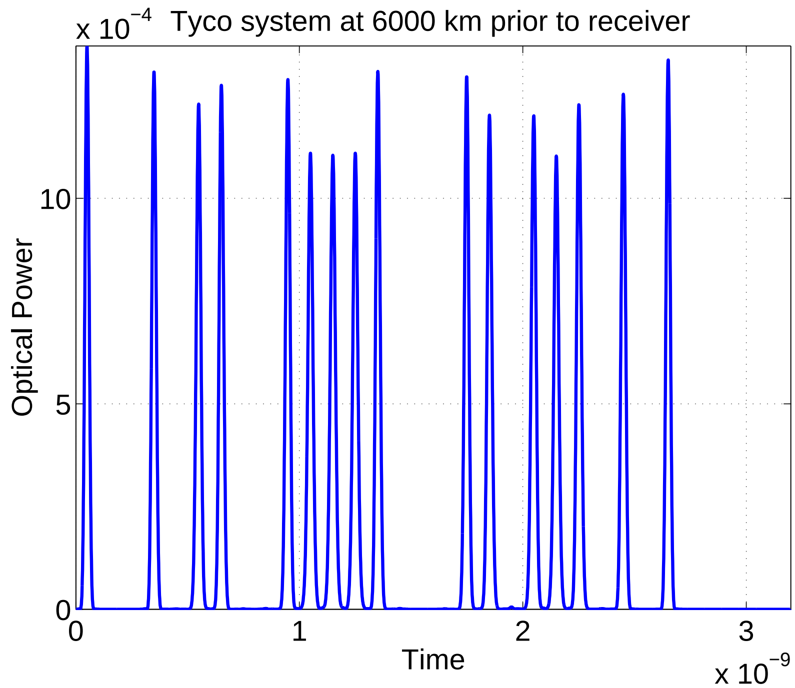

Figure 7:

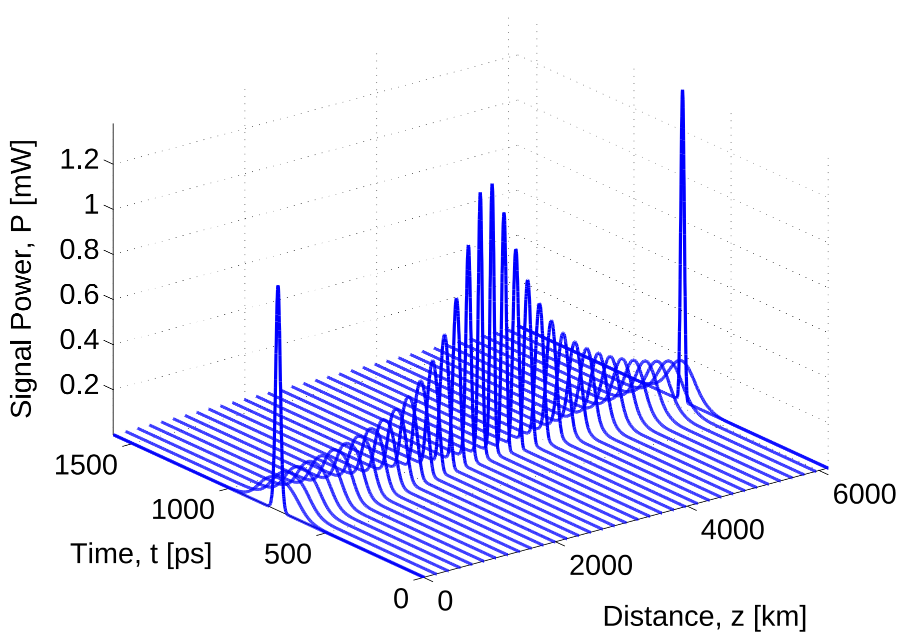

Figure 8:

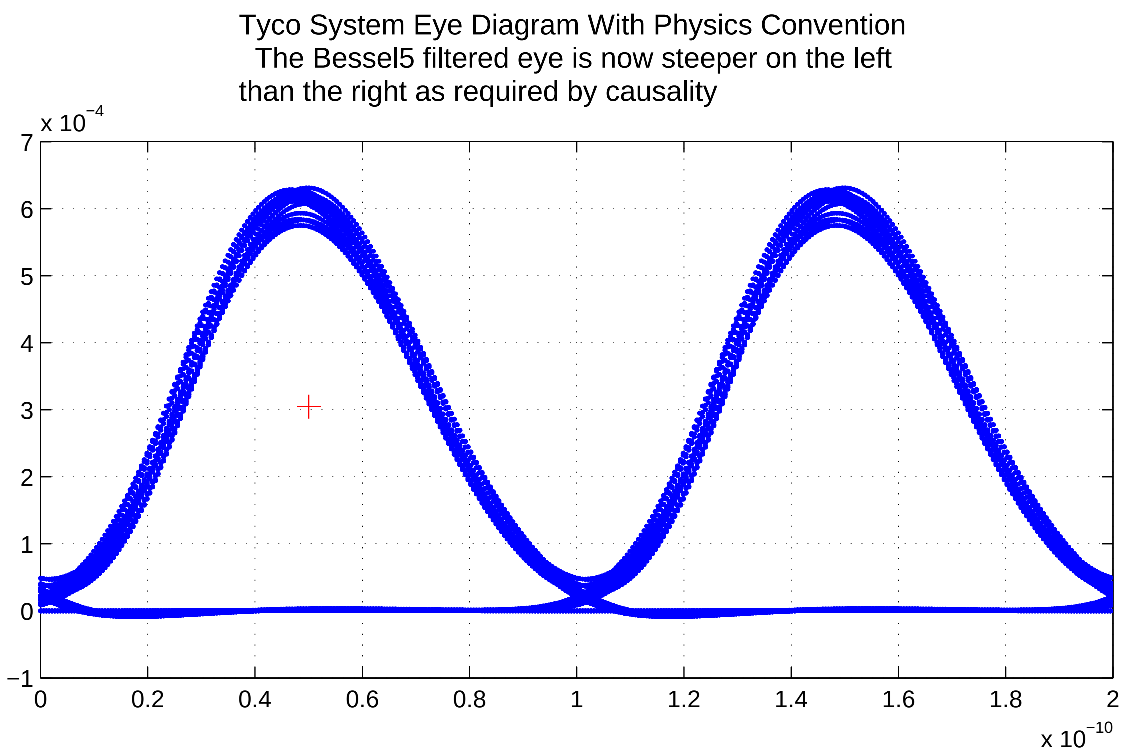

Figure 9: Tyco Sysytem Eye