
Tutorial 7: Fast Fourier Transforms in *Mathematica*

BRW
8/01/07

```
Off[General::spell];
```

This tutorial demonstrates how to perform a fast Fourier transform in *Mathematica*. The example used is the Fourier transform of a Gaussian optical pulse.

First, define some parameters. Note that all wavelength values are in nm and all time is in fs. Thus the speed of light c is 300 nm/fs and all frequencies ω is thus represented in radians/fs. The fundamental wavelength in interest, λ_o , is 800 nm with its corresponding frequency ω_o . For all values the unit for time will be femtosecond, the unit of length nanometers, and the unit for power is Watts.

```
c = 300;
λo = 800;
ωo =  $\frac{2 \pi c}{\lambda_o}$ ;
ωo // N
2.35619
```

Since a Fast Fourier Transform (FFT) is used, one must be careful to sample the electric field properly. To prevent any aliasing, the range is set such that the value of the pulse electric field is approximately zero at the ends of the range. Define the temporal step δt that the pulse electric is sampled in order to prevent aliasing. Also, the FFT requires that the number of points that sample the pulse, num , must be a power of two. In this case $\text{num}=2048$. Once δt is set, the step in the frequency domain $\delta \omega$ is determined by the equation $\delta \omega = \frac{2\pi}{\text{num} \delta t}$. The value range is the temporal range in fs over which the electric field is defined.

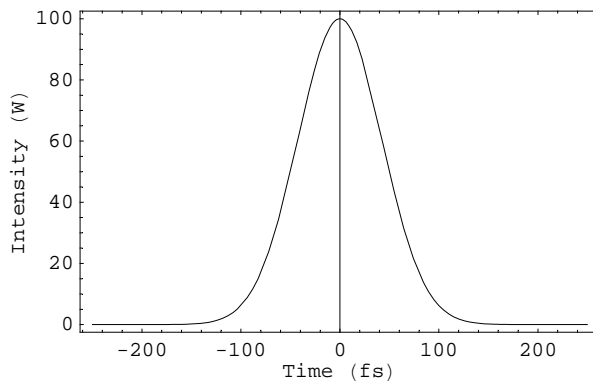
```
num = 212
range = 2000; (* time in femtoseconds *)
δt = range / num // N
δω =  $\frac{2 \pi}{\text{num} \delta t}$  // N
4096
0.488281
0.00314159
```

For the FFT it will be helpful to generate areas of time, relative frequency, and absolute frequency.

```
time = Table[(j - (num / 2)) * δt, {j, 1, num}];
freq = Table[(j - (num / 2)) * δω, {j, 1, num}];
absfreq = Table[freq[[j]] + ωo, {j, 1, num}];
```

Define electric field temporal envelope $e[t]$ as a Gaussian. Its intensity will be $I_0[t,w]$. Note that w is the full-width at half maxima (FWHM) of the intensity (not field).

```
w = 100; (* pulse full width at half maximum in femtoseconds *)
Pave = 0.001; (* average power in W *)
T = (1 / (100 106)) 1015; (* pulse repetition period in femtoseconds *)
Po = Pave T / w; (* peak power in W *)
e[t_] = Sqrt[Po] Exp[-2 Log[2] (t/w)2];
I0[t_] = e[t] * Conjugate[e[t]];
Plot[I0[t], {t, -250, 250}, Frame -> True,
  PlotRange -> {All, All}, FrameLabel -> {"Time (fs)", "Intensity (W)"}];
```



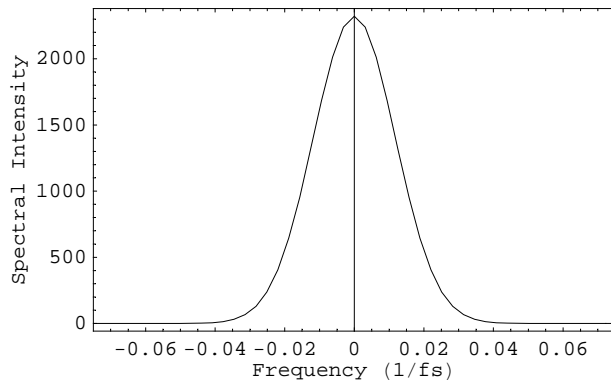
Let's compute the spectrum of the Gaussian pulse using the Fourier transform. I will now define a specific notation. Any variable with the word "data" will be an array (or list as known in Mathematica) of values. Anything with the generic form "f[]" is a function. To use the FFT, the function $e[t]$ is sampled and represented by variable $etdata$. However, the input string to the FFT must have the $t=0$ element as the first element in the string. This is done by the `RotateLeft` function and saved as $nydata$. $etdata$ is plotted again to compare with the above curve. The intensity of the electric field will have a temporal pulse width of 100 fs.

```
etdata = Table[e[time[[j]]], {j, 1, num}];
nydata = RotateLeft[etdata, num / 2 - 1];
ewdata = Chop[RotateRight[Fourier[nydata], num / 2 - 1]];
Iwdata = ewdata * Conjugate[ewdata];
```

The "RotateRight" is used to place the zero frequency component in the middle of the data string for plotting. The "Chop" command replaces approximate real numbers in $expr$ that are close to zero by the exact integer 0.

Plot the spectrum in frequency

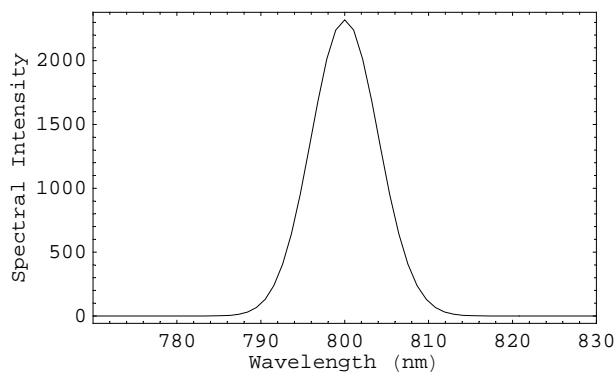
```
ListPlot[Table[{freq[[j]], Iωdata[[j]]}, {j, 1, num}], Frame -> True,
PlotRange -> {{-0.075, 0.075}, All}, PlotJoined -> True,
FrameLabel -> {"Frequency (1/fs)", "Spectral Intensity"}];
```



Plot the spectrum in wavelength, we need to define a wavelength array.

```
zp = Position[absfreq, _? (-0.001 < # < 0.001 &)] [[1, 1]] + 1;
wave = Reverse[Table[ $\frac{2 \pi c}{\text{absfreq}[[i]]}$ , {i, zp, Length[freq]}]];
eλdata = Reverse[Table[eωdata[[i]], {i, zp, Length[freq]}]];
Iλdata = eλdata * Conjugate[eλdata];

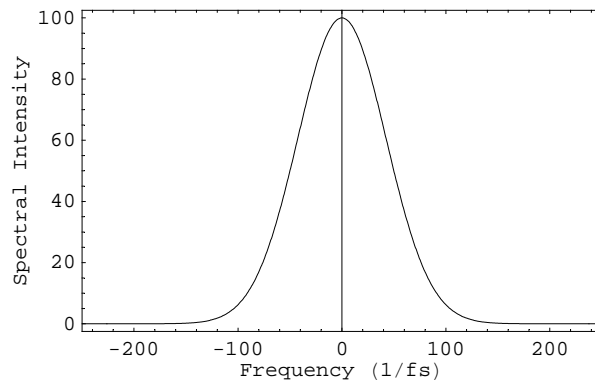
ListPlot[Table[{wave[[j]], Iλdata[[j]]}, {j, 1, Length[wave]}], Frame -> True,
PlotRange -> {{770, 830}, All}, PlotJoined -> True,
FrameLabel -> {"Wavelength (nm)", "Spectral Intensity"}];
```



Now, lets take the inverse Fourier transform of $E(\omega)$ to get the original temporal electric field $E(t)$. Plot the corresponding temporal intensity.

```
nydata = RotateLeft[eωdata, num / 2 - 1];
et2data = Chop[RotateRight[InverseFourier[nydata], num / 2 - 1]];
It2data = et2data * Conjugate[et2data];
```

```
ListPlot[Table[{time[[j]], It2data[[j]]}, {j, 1, num}], Frame -> True,  
PlotRange -> {{-250, 250}, All}, PlotJoined -> True,  
FrameLabel -> {"Frequency (1/fs)", "Temporal Intensity"}];
```



This is the same intensity we started off with.