$'$
$_explained.tex`andcompilewith`pdflatex`(orOverleaf).$

Inverse FFT (IFFT) Pipeline for Transform-Limited Pulse Retrieval

A Mathematical Walkthrough Aligned with the MATLAB Script

Overview

This note formalizes the inverse-Fourier pipeline used to transform a measured spectrum $I(\lambda)$ (wavelength domain) into an estimated time-domain intensity $I(t) = |E(t)|^2$ under a transform-limited (zero spectral phase) assumption. It follows the steps implemented in the MATLAB script: background removal, normalization, interpolation to a uniform frequency grid, IFFT to time, centering, and optional forward FFT back to spectrum.

1 From Wavelength to Frequency

Let c denote the speed of light in vacuum ($c = 299\,792\,458\,\mathrm{m/s}$). The relations between wavelength and frequency are

$$f = \frac{c}{\lambda}, \qquad \lambda = \frac{c}{f}. \tag{1}$$

The input file provides $I(\lambda)$ (typically λ in nm). If the file stores λ in meters, the script converts to nm for plotting, but always uses meters when computing f.

Background Removal and Normalization. Choose a background window $[\lambda_a, \lambda_b]$ (e.g. 700 nm–750 nm) and estimate a baseline I_{bk} (mean) and its standard deviation σ_{bk}. Define the background-corrected intensity

$$I_{\mathrm{corr}}(\lambda) = \max\left(I(\lambda) - I_{\mathrm{bk}},\, 0\right), \tag{2}$$

and optionally zero out points with $I(\lambda) \le I_{\mathrm{bk}} + k\,\sigma_{\mathrm{bk}}$ (with $k \approx 2$) to suppress near-baseline noise. Then normalize to unit peak:

$$\tilde{I}(\lambda) = \frac{I_{\mathrm{corr}}(\lambda)}{\max_\lambda I_{\mathrm{corr}}(\lambda)} \in [0, 1]. \tag{3}$$

Mapping to frequency via $f = c/\lambda$ yields $\tilde{I}(f)$.

2 Spectral Field Amplitude (Transform-Limited Assumption)

Assuming zero spectral phase (transform-limited pulse), the spectral *field amplitude* magnitude is taken as

$$E(f) = \sqrt{\tilde{I}(f)} \quad \text{(real, nonnegative)}. \tag{4}$$

No phase information is introduced; therefore, the time-domain field obtained by IFFT represents a transform-limited estimate consistent with the measured spectral *magnitude* only.

3 Uniform Frequency Grid and Interpolation

Fast Fourier transforms require uniform sampling. Let the measured positive-frequency band be $f \in [f_{\min}, f_{\max}]$ with samples $\{(f_i, E(f_i))\}$. Construct a uniform grid

$$f_k = f_{\min} + k\,\Delta f, \qquad k = 0, 1, \ldots, N_+ - 1, \tag{5}$$

and, for convenience in the script, extend to a symmetric grid

$$\hat{f}_k \in [-f_{\max},\, f_{\max}], \qquad k = 0, 1, \ldots, N - 1, \tag{6}$$

with constant spacing Δf. Interpolate $E(f)$ onto the uniform grid where data exist, and set it to zero outside $[f_{\min}, f_{\max}]$:

$$\hat{E}(\hat{f}_k) = \begin{cases} \text{interp}[E(f)]\,, & \hat{f}_k \in [f_{\min}, f_{\max}], \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Note. Because only the magnitude is used and no Hermitian symmetry is enforced, $\hat{E}$ is generally not conjugate-symmetric. The resulting time-domain field may be complex and can exhibit small artifacts; this is expected when phase is unknown.

4 Inverse Fourier Transform to Time Domain

The continuous inverse Fourier transform (frequency $\leftrightarrow$ time, with 2π in the exponent) is

$$E(t) = \int_{-\infty}^{\infty} E(f)\, e^{i2\pi f t}\, df, \qquad I(t) = |E(t)|^2. \tag{8}$$

On the discrete, uniformly sampled grid $\{\hat{f}_k\}$, MATLAB's `ifft` approximates this integral via

$$E[n] = \frac{1}{N} \sum_{k=0}^{N-1} \hat{E}[\hat{f}_k]\, e^{i2\pi \frac{kn}{N}}, \qquad n = 0, 1, \ldots, N - 1, \tag{9}$$

where N is the total number of frequency samples. The frequency and time sampling are related by

$$\Delta t = \frac{1}{N\,\Delta f}, \qquad T = N\,\Delta t = \frac{1}{\Delta f}, \tag{10}$$

so that the discrete time vector is

$$t_n = n\,\Delta t, \qquad n = 0, 1, \ldots, N - 1, \tag{11}$$

and the time-domain intensity is

$$I(t_n) = |E[n]|^2. \tag{12}$$

Centering the Pulse. The script finds $\arg\max_n I(t_n)$ and circularly shifts $I(t)$ so that the main peak is centered (i.e. near $t = 0$). This changes only the time reference, not the pulse shape.

5 Optional: Forward FFT Back to Spectrum

For completeness, the forward transform (discrete) is

$$\tilde{E}[k] = \sum_{n=0}^{N-1} E[n]\, e^{-i2\pi \frac{kn}{N}}, \qquad k = 0, 1, \ldots, N-1, \tag{13}$$

from which a reconstructed spectral intensity can be formed as

$$I_{\mathrm{rec}}(f_k) = \left|\tilde{E}[k]\right|^2. \tag{14}$$

Mapping $f_k \mapsto \lambda_k = c/f_k$ yields $I_{\mathrm{rec}}(\lambda)$ for comparison with the input envelope (monotonicity in λ is not guaranteed unless frequencies are sorted).

6 Bandwidth and Pulse Duration Notes

Super-Gaussian in Wavelength

A commonly used model for the spectral envelope is the super-Gaussian

$$I(\lambda) = A\, \exp\left(-|\frac{\lambda - \lambda_0}{C}|^{2n}\right) + D, \tag{15}$$

with amplitude A, center λ_0, scale C, order $n \geq 1$, and offset D. Its full width at half maximum (FWHM) is

$$\mathrm{FWHM}_\lambda = 2\,C\,(\ln 2)^{1/(2n)}. \tag{16}$$

The script prints $\approx 1.825\,C$, which is exact when $n = 2$.

Gaussian Time-Domain Fit

The time-domain intensity is fitted near the peak by a Gaussian model

$$I(t) = A\, \exp\left[-2\left(\frac{t}{C}\right)^2\right], \tag{17}$$

for which the FWHM in time is

$$\mathrm{FWHM}_t = C\,\sqrt{2\ln 2}. \tag{18}$$

This yields a transform-limited estimate of the pulse duration from the measured spectrum *magnitude*.

7 Algorithm Summary (as Implemented)

1. Load (λ, I). Convert units if needed. Plot raw spectrum.
2. Estimate background on a chosen window; subtract DC and zero near-baseline values.
3. Normalize to unit peak; map to frequency using $f = c/\lambda$.
4. Form spectral field magnitude $E(f) = \sqrt{\tilde{I}(f)}$.
5. Interpolate $E(f)$ onto a uniform grid $\hat{f}_k \in [-f_{\max}, f_{\max}]$; set outside-band values to zero.
6. Compute $E[n] = \texttt{ifft}(\hat{E}[\hat{f}_k])$; obtain $I(t_n) = |E[n]|^2$, with $t_n = n/(N\Delta f)$.
7. Center the pulse by circular shift. Optionally filter a tiny window around $t = 0$.
8. (Optional) Apply `fft` to return to spectrum and compare envelopes.
9. Fit super-Gaussian in λ (input and reconstructed); report λ_0 and FWHM.
10. Fit Gaussian in time near the peak; report FWHM_t.
11. Save spectra and time traces with headers and units.

8 Caveats

- No spectral phase $\Rightarrow$ transform-limited assumption; real pulses with chirp will be longer.
- Not enforcing Hermitian symmetry on $\hat{E}$ can yield complex $E(t)$ and small artifacts; this is a standard consequence of unknown phase.
- The wavelength-domain reconstruction from discrete f_k may be non-monotonic in λ; use it primarily to compare *envelopes*.
- Background window and threshold $(\lambda_a, \lambda_b, k)$ affect the recovered bandwidth and thus the TL duration.

Symbols and Units

Symbol	Meaning / Unit
c	Speed of light, $299\,792\,458\,\mathrm{m/s}$
λ	Wavelength (input), usually nm (converted to m for computing f)
f	Frequency (Hz), $f = c/\lambda$
$I(\lambda)$, $\tilde{I}(\lambda)$	Measured & normalized spectral intensity (arb. units)
$E(f)$	Spectral field magnitude, $E(f) = \sqrt{\tilde{I}(f)}$
Δf, Δt	Frequency & time sampling steps, $\Delta t = \frac{1}{N\Delta f}$
$E[n]$, $I(t_n)$	Discrete time-domain field and intensity, $I(t_n) = \lvert E[n]\rvert^2$
FWHM_λ	Spectral FWHM (super-Gaussian: $2C(\ln 2)^{1/(2n)}$)
FWHM_t	Temporal FWHM (Gaussian: $C\sqrt{2\ln 2}$)

Tip. If you want to enforce a real-valued $E(t)$, mirror the positive-frequency magnitude and (optionally) set a zero phase to impose conjugate symmetry: $\hat{E}(-f) = \hat{E}(f)$. This changes details of the waveform but remains consistent with the transform-limited assumption.