Function Guide for the Fourier Transformation Package
SPIRE-UOL-DOC-002496

Prepared by:

Peter Davis (University of Lethbridge)
peter.davis@uleth.ca

Andres Rebolledo (University of Lethbridge)
andres.rebolledo@uleth.ca

Trevor Fulton (University of Lethbridge)
trevor.fulton@uleth.ca

Approved by:

David Naylor (University of Lethbridge)
naylor@uleth.ca

Document History:

Issue	Date
Version 1.0	**25 July 2005**
Version 1.1	**17 August 2005**

Acronyms

HKT	HouseKeeping Timeline
OPD	Optical Path Difference
RMS	Root Mean Square
SMECT	Spectrometer Mechanism Timeline
SDI	Spectrometer Detector Interferogram
SDS	Spectrometer Detector Spectrum
SDT	Spectrometer Detector Timeline
ZPD	Zero Path Difference

Table of Contents

1 Purpose of this document

The purpose of this document is to introduce the Fourier Transform software package. The individual functions that make up the Fourier Transform package will be presented in terms of the effect they have on data from spectrometer observations and how they relate to one another.

2 The Fourier Transform Package

The main purpose of the Fourier Transform package is to convert the data collected from a spectrometer observation into a set of spectral products. At a minimum, the processing steps required to accomplish this goal consist of two functions; one function to create a set of interferograms, the other function to transform these interferograms into spectra (see Figure 1).

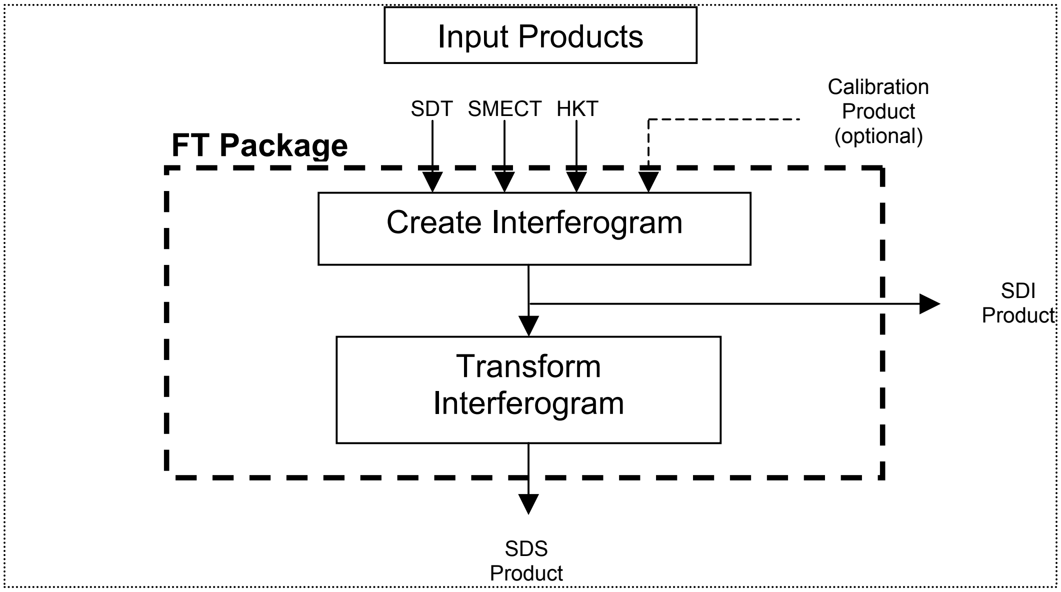

Figure 1: Basic functionality of the Fourier Transform package

Owing to procedural nature of the processing required, the Fourier Transform package has been broken up into a set of atomic functions, or tasks. This choice was made to allow for a degree of flexibility in the manner by which data from spectrometer observations are processed.

An overview of the individual tasks of the Fourier Transform package and their connection with one another is shown in Figure 2 while descriptions of these subtasks are presented in §2.1—§2.6. The *SPIRE Pipeline Description*[1] document describes how the Fourier Transform package fits within the overall SPIRE Spectrometer data processing pipeline.

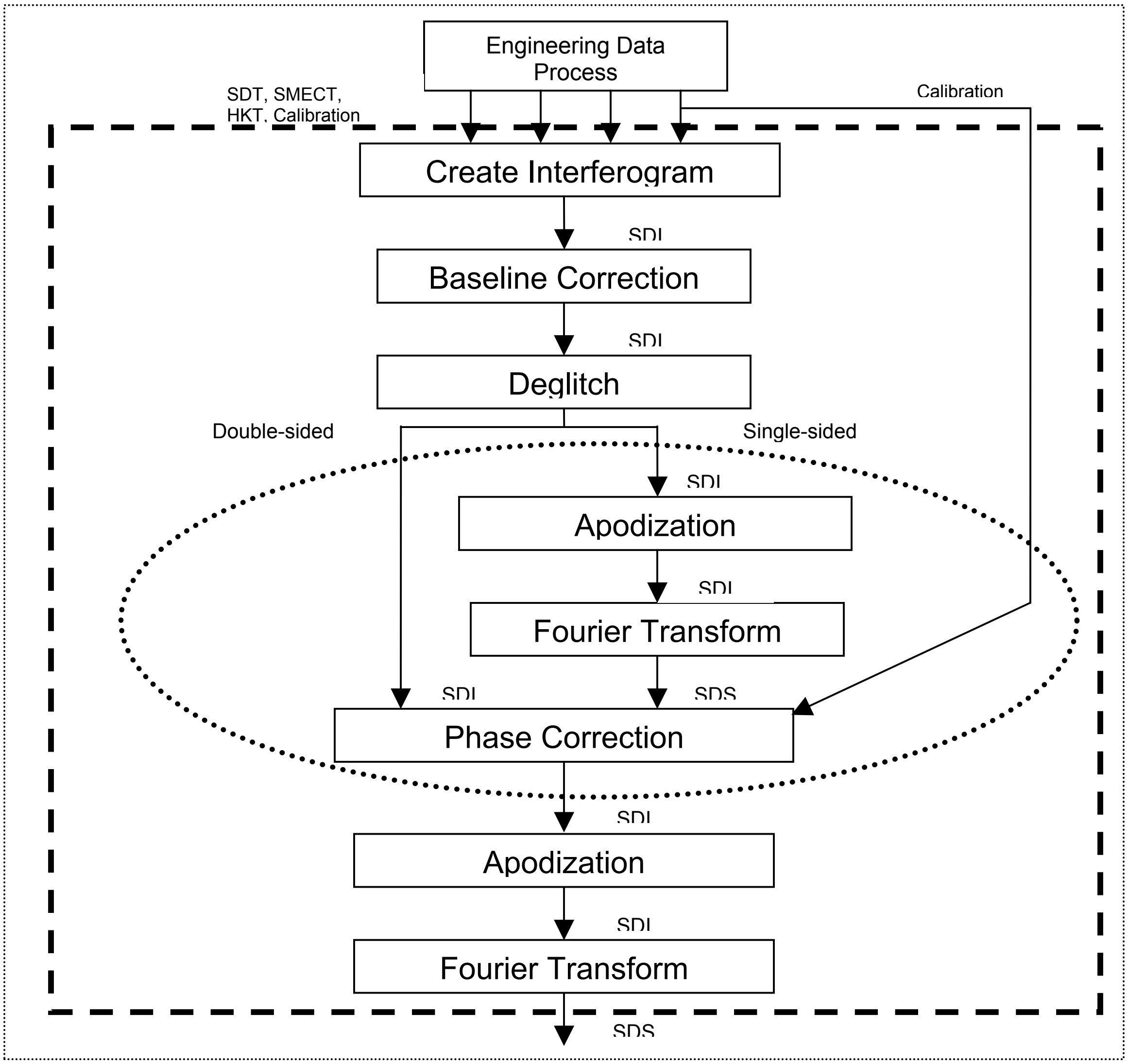

Figure 2: Detailed functionality of the Fourier Transform package. Subtasks and their input and output products are shown.

---

[1] Tanya Lim, "*SPIRE Pipeline Description*", **SPIRE-RAL-DOC-002437**, **Draft 0.1**, page 26, 28, 18 May 2005.

2.1 Create Interferograms

Purpose: This step creates a set of interferograms (i.e. detector signal as a function of optical path difference) for a given spectrometer observation.

Description: The initial data products recorded during SPIRE spectrometer observations consist of a spectrometer detector timeline (SDT) and a spectrometer mechanism timeline (SMECT). In order to calculate a spectrum from these data it is first necessary to combine the data from these timelines in such a way as to link the recorded detector signal, I, to the position of the spectrometer mechanism, x, thereby creating an interferogram, I(x). Moreover, in the current implementation of the Fourier Transform package, it is required that the spacing between samples of the interferogram be uniform so that the Fast Fourier Transform can be used to compute the desired spectrum.

Detector signals I(t') from the input SDT product and irregularly-gridded stage positions z(t) from the input SMECT Product are interpolated to create a new regularly-gridded interferogram Product, I'(x). The interpolation process consists of two steps; the first interpolates the irregularly-gridded SMEC position timeline z(t) to a regularly-gridded SMEC position timeline x(t''), the second step interpolates the measured detector signal I(t') to the regularly-gridded SMEC positions I(t'') (=I(x)). In addition, a calibration product may be used to register the regularly sampled position grid, x, to the position of zero optical path difference (ZPD). If no calibration product is specified, a default value denoting the ZPD position is used. In its current implementation, this function uses a cubic spline to perform each interpolation.

Task: `RegSampledIfgmCreation`.
Mandatory Input Products: SDT, SMECT, HKT.
Optional Input Products: Calibration product containing the position of zero optical path difference for each detector pixel.
Output Products: SDI.
Future Development:

- It may be possible to significantly improve on the signal interpolation and the stage interpolation. A windowed-sinc interpolation scheme would be preferable. However, problems due to the irregular sampling of the data make the windowed-sinc interpolation more difficult to implement.

2.2 Fourier Transform

Purpose: This step creates a set of spectra (i.e. intensity as a function of wavenumber) for each spectrometer detector for a given observation. This function may be used to transform either single-sided or double-sided interferograms.

Description: The process by which the input interferogram product is transformed to a spectral product is by way of the Fast Fourier Transform (FFT). As such, this requires that the interferograms in the input SDI product be sampled on a regular optical path difference grid. The Fourier Transform function is able to transform either single-sided or double-sided interferograms, as specified by a keyword input. If the input interferogram is specified as double-sided then the calculated spectrum contains both real and imaginary components. If the input interferogram is single-sided, only the portion of the interferogram whose optical path difference is greater than or equal to zero is used to calculate a spectrum and the resultant spectrum is entirely real.

Task: `RegSampledFourierTransform`.
Mandatory Input Products: SDI.
Optional Input Products: None.
Output Products: SDS.

2.3 Phase Correction

Purpose: This function corrects the interferograms in the input SDI product for any phase that may be caused missampling of the position of zero path difference or by the presence of dispersive elements in the spectrometer (e.g. optics, electronics).

Description: The even symmetry of the spectrometer theoretically implies that its calculated spectrum should be a real function, as shown in the equation below.

$$FT(I_{EVEN}(x)) = B(\sigma) = B_{\mathrm{Re}}(\sigma) \qquad \mathbf{(1)}$$

The presence of dispersive elements and particularly the possibility that the position of zero path difference is not properly sampled leads to a loss of this even symmetry and a resultant spectrum that contains both real and imaginary components.

$$\begin{aligned} FT(I(x)) &= B_{\mathrm{Re}}(\sigma) + iB_{\mathrm{Im}}(\sigma) \\ &= B(\sigma)e^{i\varphi(\sigma)} \end{aligned} \qquad \mathbf{(2)}$$

An example of a spectrum containing both real and imaginary components is shown in Figure 3.

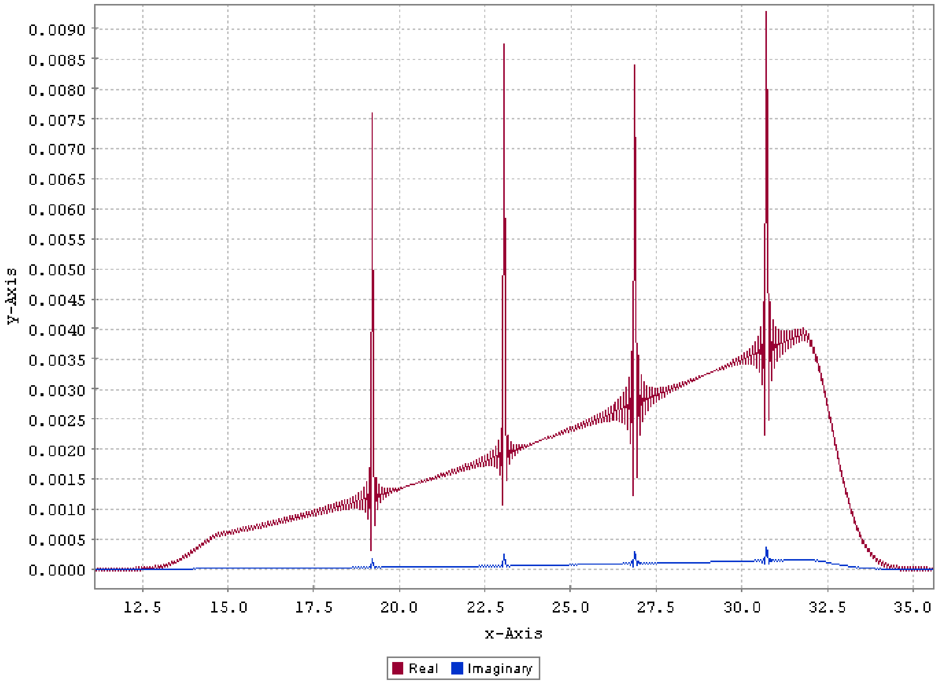

Figure 3: Fourier Transform of an uneven double-sided interferogram. The **red** curve is the real portion of the spectrum; the **blue** curve is the imaginary portion of the spectrum.

The process of phase correction moves the components of the spectrum that are located in the imaginary domain to the real domain.

Much like the Fourier Transform function described in §2.2, the Phase Correction function can be used to correct either single-sided or double-sided interferograms.

Double-sided Phase Correction

If the interferograms in the input SDI product are double-sided, i.e. symmetric about the ZPD position, then their resultant spectra will contain phase information for each spectral element. As such, phase correction can take place solely in the spectral domain. Referring to the equation 2 above, the calculated spectrum, $B(\sigma)e^{i\varphi(\sigma)}$, can be corrected by way of multiplication with a phase correction function (PCF) of the form $PCF = e^{-i\varphi(\sigma)}$. That is,

$$\begin{aligned}\text{Phase Corrected Spectrum} &= B(\sigma)e^{i\varphi(\sigma)} \times PCF \\ &= B(\sigma)e^{i\varphi(\sigma)} \times e^{-i\varphi(\sigma)} \\ &= B(\sigma)\end{aligned} \qquad \textbf{(3)}$$

Note that the PCF here is not simply equal to the negative of the measured phase. Rather, a fit is made to the in-band phase of the calculated spectrum and it is the negative of this fit that is used as the PCF (see Figure 4). The basis for correcting with the fitted phase rather than the calculated phase it that by doing so half of the noise associated with the measurement is left in the imaginary domain. For random sources of measurement noise, this can lead to an increase in the signal-to-noise ratio by a factor of $\sqrt{2}$.

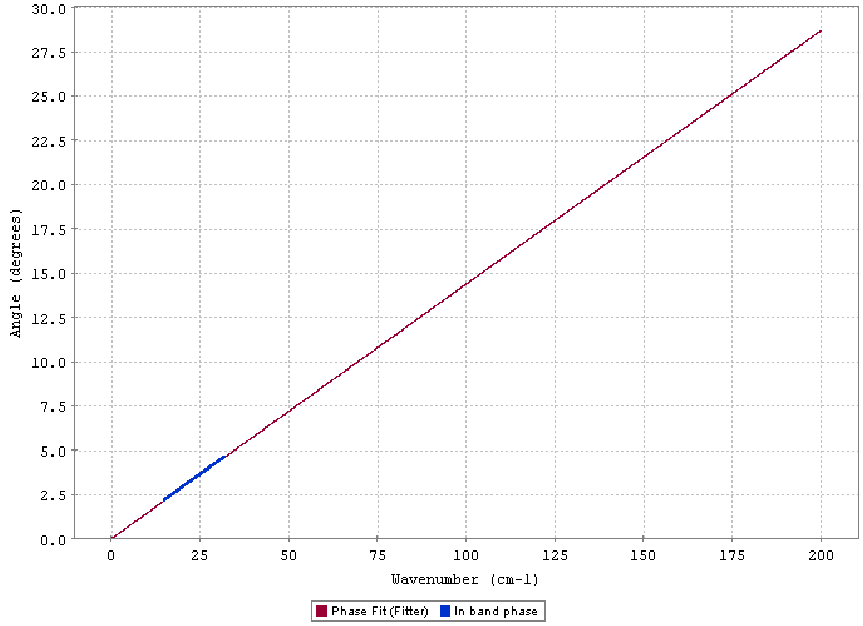

Figure 4: Fitting a function to the measured phase. The **blue** curve is the spectral phase derived from the double-sided interferogram; the **red** curve is a fitted function to the in-band portion of the measured phase.

Single-sided Phase Correction

Phase correction for single-sided interferograms, interferograms that are not symmetric about the ZPD position (see Figure 5), while mathematically equivalent to phase correction of double-sided interferograms, differs slightly from a procedural point of view. A different procedure must be employed because, for single-sided interferograms, the uncorrected spectrum (magnitude and phase) can only be calculated for the lower resolution, symmetric portion of the interferogram.

$$\begin{aligned} FT(I_{DS}(x)) &= B_{\mathrm{Re}}(\sigma) + iB_{\mathrm{Im}}(\sigma) \\ &= B_{DS}(\sigma)e^{i\varphi(\sigma)} \end{aligned} \qquad \textbf{(4)}$$

The phase correction function computed from the symmetric portion of the single-sided interferogram can still be used to correct the single-sided interferogram, however. Recall that for Fourier Transforms, multiplication in one domain is equivalent to convolution in the other domain. Thus, rather than multiplying the single-sided spectrum by a PCF, phase correction can be achieved by convolving the single-sided interferogram with the inverse transform of the PCF. That is,

$$\begin{aligned} \begin{matrix}\text{Single sided} \\ \text{PhaseCorrection}\end{matrix} &= I(x) \otimes FT^{-1}(PCF) \\ &= I(x) \otimes FT^{-1}(e^{-i\varphi(\sigma)}) \end{aligned} \qquad \textbf{(5)}$$

where the PCF is found from the phase of the symmetric portion of the single-sided interferogram (see Figure 6 and Figure 7). The result of the convolution given in equation 6 is the single-sided interferogram is rendered symmetric about its position of zero path difference. Put another way, the single-sided interferogram is converted to an even function (see Figure 8). As such, only the portion of the single-sided interferogram where the optical path difference is greater or equal to zero needs to be used to compute the resultant spectrum (see Figure 9).

Task: `RegSampledPhaseCorrection`.
Mandatory Input Products: SDI.
Optional Input Products: None.
Output Products: SDI.
Future Development:

- Eventually, it is hoped that through calibration all of the possible systematic sources of phase will be known *a priori*. Phase correction will then be a matter of multiplication (or convolution) with a known phase function (given in a calibration product) rather than using a function derived from a fit to the given observation data.

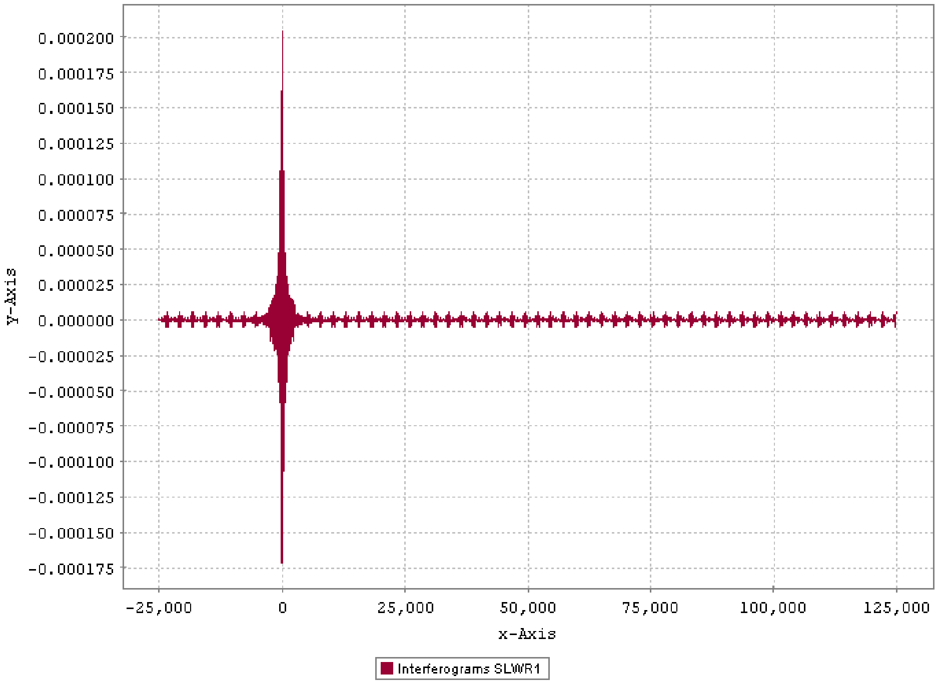

Figure 5: Single-sided interferogram. Note the short symmetric portion about ZPD (x=0).

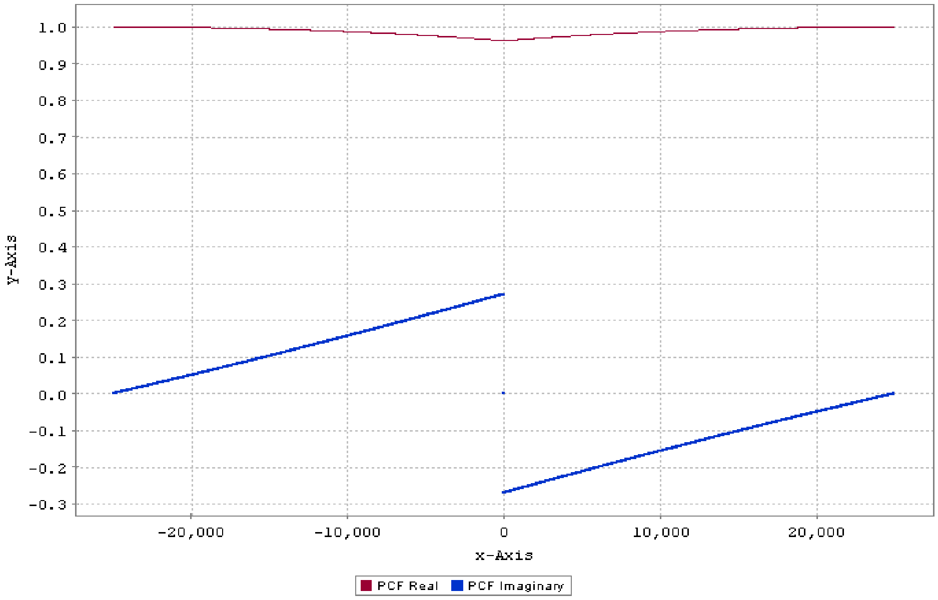

Figure 6: Phase correction function, $e^{-i\varphi(\sigma)}$. The **red** curve is the real portion of the PCF; the **blue** curve is the imaginary portion of the PCF.

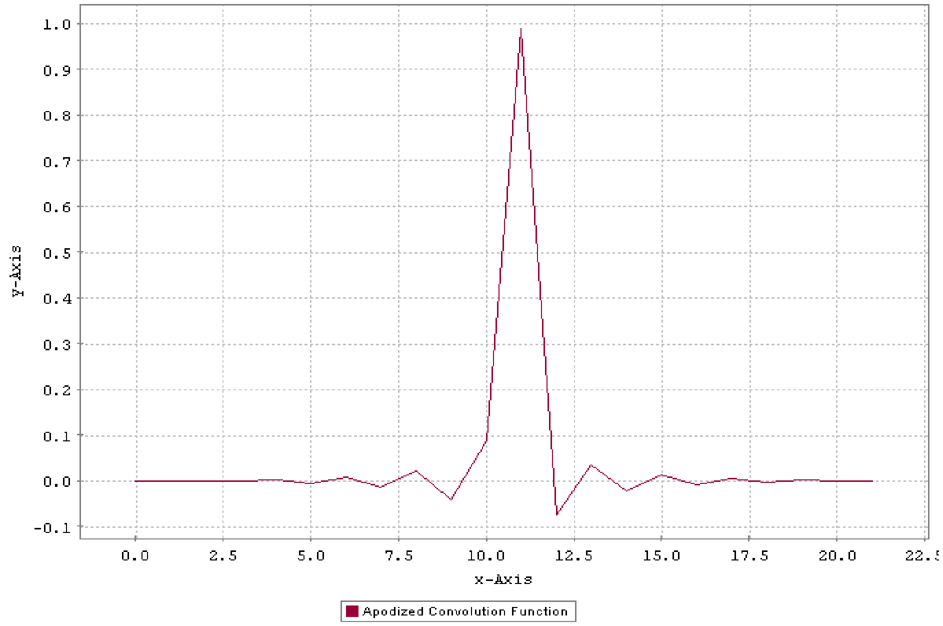

Figure 7: Inverse transform of the phase correction function. Phase correction of the single-sided interferogram is achieved by convolution with this function.

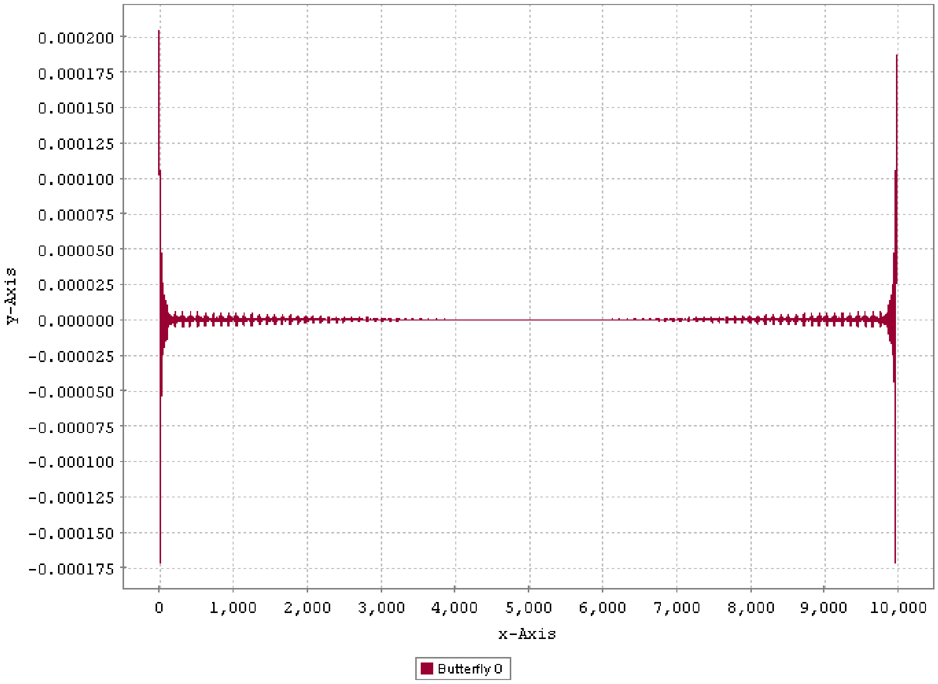

Figure 8: Apodized (NB 1.9 FWHM), phase corrected, and butterflied single-sided interferogram

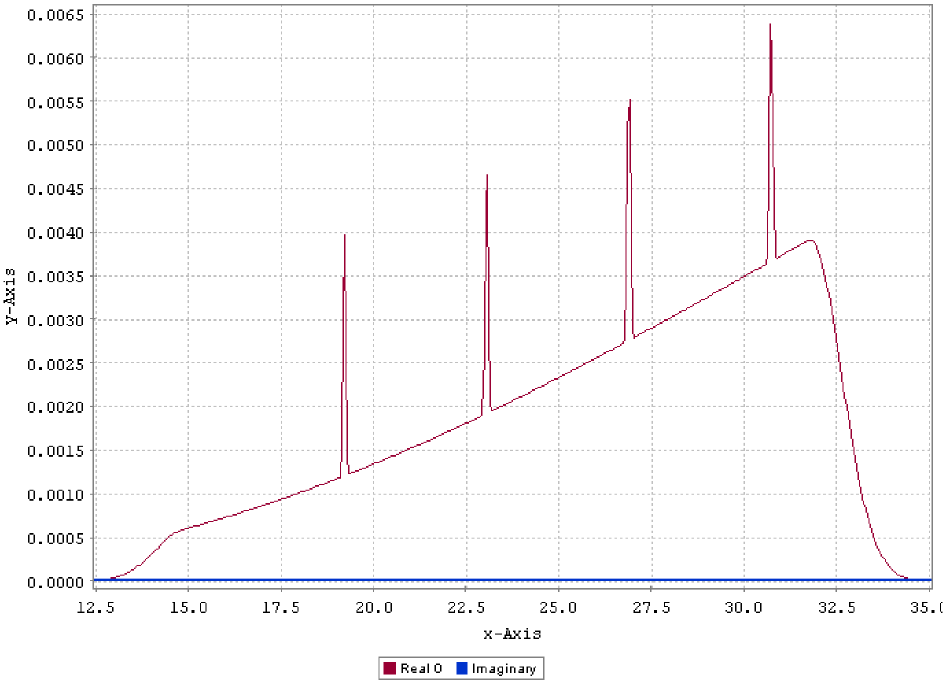

Figure 9: Spectrum from a phase corrected single-sided interferogram.

2.4 Baseline Correction

Purpose: Correct the baseline of the measured interferogram for drifts.

Description: Baseline correction removes any offset and drift in the baseline of the measured interferogram. This function is important because failure to remove any offset or drift can lead to artifacts in the calculated spectrum. This baseline correction function should be used if the intention is to zero-pad the interferograms prior to transformation.

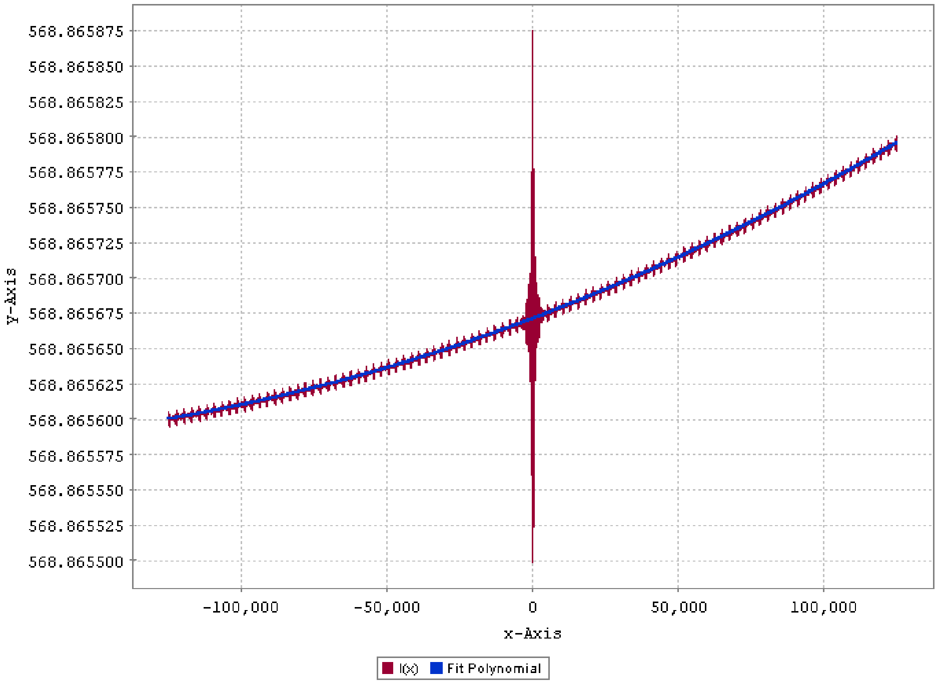

Figure 10: Interferogram with simulated 2nd degree polynomial drift. The blue curve is a fitted function to the baseline of the interferogram.

Task: `DriftRemoval`.
Mandatory Input Products: SDI.
Optional Input Products: None.
Output Products: SDI.
Future Development:

- Baseline correction may be more properly performed on the SDT product rather than the SDI product.

2.5 Deglitching

Purpose: To remove unwanted localized artifacts from the measured interferogram prior to transformation.

Description: Glitches in the signal measured with a bolometer can arise from sources such as the impact of an ionizing particle, instrument microphonics, or disturbances in the laboratory/spacecraft environment. Glitches pose a serious problem for spectrometer observations since a glitch affecting as few as a single point in an interferogram can affect every point in the spectrum. Moreover, it is not sufficient to simply identify a glitch in an interferogram, the glitch must also be removed and the affected points must be replaced.

The deglitching function in the Fourier Transform package is itself broken into two steps; the first step scans the set of interferograms in an effort to identify any glitches, the second step removes and replaces the glitches.

One of the properties of the SDI products created by the Fourier Transform package is that the samples for each interferogram for a given detector pixel are all on the same position grid. As such, glitches can be detected by comparing the measured signal of each interferogram scan for a given pixel at each OPD position. Glitches are identified by flagging outliers of the scan-to-scan comparison at each sample point using metrics such as standard deviation or skewness (see Figure 11 and Figure 12). In general, standard deviation is more accurate for observations with a low number of scans (# of scans < 10), while the skewness technique works better for observations with a large number of scans (# of scans ≥ 10).

After identifying the glitches present in the interferograms of the SDI product, the offending points are removed and then replaced. The values of the replacement points are taken as the average of the values from the unaffected interferograms at that OPD position (see Figure 13).

Task: `IfgmDeglitcher`.
Mandatory Input Products: SDI.
Optional Input Products: None.
Output Products: SDI.

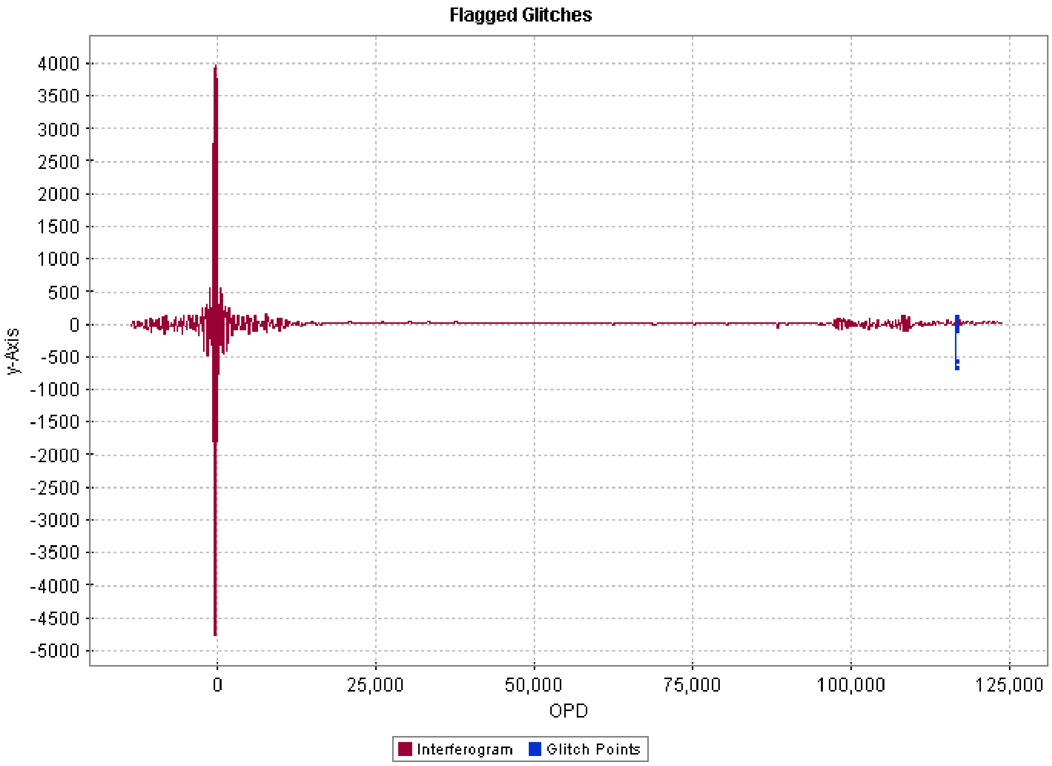

Figure 11: Interferogram with simulated glitch. The points shown in **blue** have been flagged for correction.

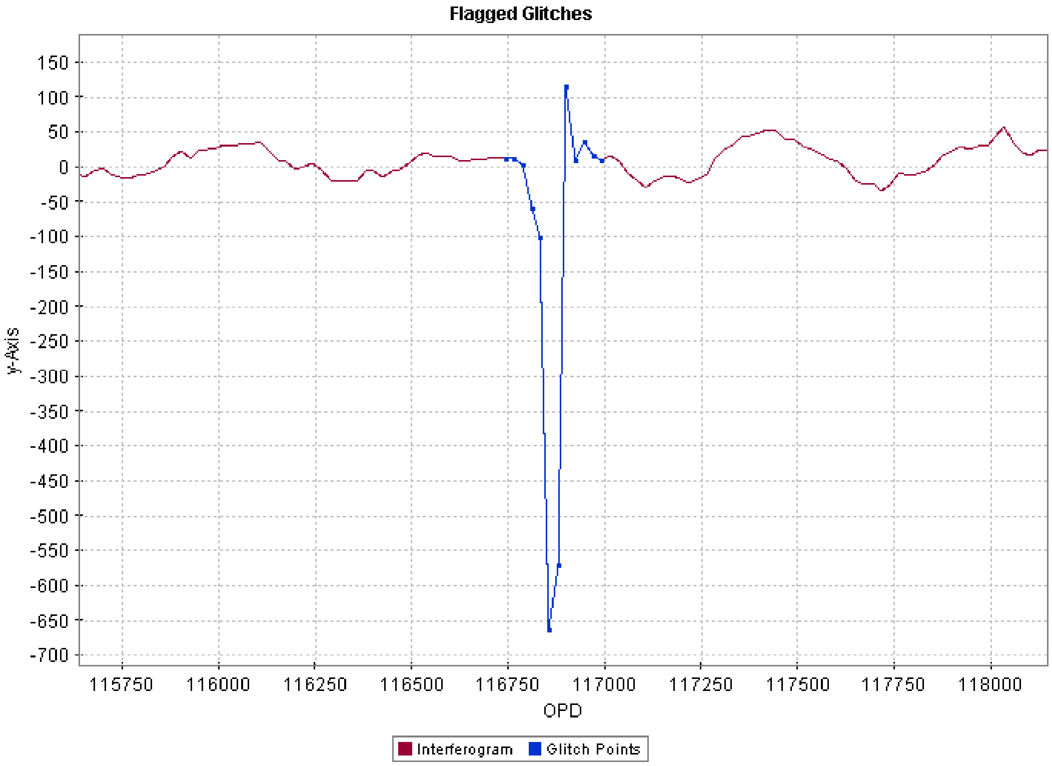

Figure 12: Interferogram with simulated glitch. Close-up version of the plot shown in **Figure 11**. The points shown in **blue** have been flagged for correction.

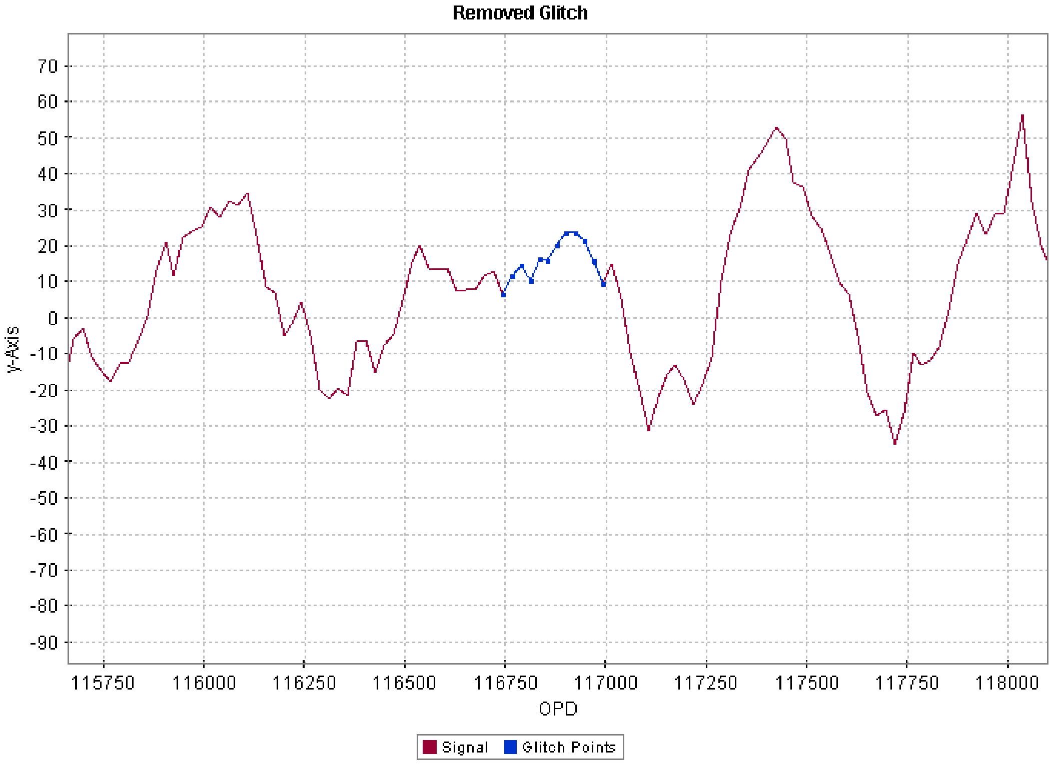

Figure 13: Interferogram with the glitch points removed and replaced. The points shown in **blue** have been corrected.

Future Development:

- Tweaking of the algorithms is TBD. One possible change suggested in the standard deviation deglitching is to scan a window function across the timeline and flag those points that fall outside the window. This method may account for variations in signal standard deviation in the wing region versus the central peak region.
- It is still not clear in which situations is the skewness glitch detection method preferable to the standard deviation method. The skewness deglitching algorithm has been used successfully to deglitch data containing a large number of interferograms (as was the case for the CQM2 data). However, tests on simulated data using small numbers of datasets (< 10 interferograms) have shown the skewness algorithm to be too sensitive to the point where it flags valid points as glitches.
- Cross-scan detections methods to flag outliers are still in development. Skewness and standard deviation provide an outlier threshold, which is constant across an entire scan. The drawback of this method is that the level of modulation is different in the central and wing regions of the interferograms, making detection of glitches more difficult in one or the other region. A detection method that uses median per-point threshold has the potential to accurately identify glitches in both the central and wing regions.

2.6 Apodization

Purpose: Remove spectral artifacts related to the instrument line shape of the spectrometer.

Description: The natural instrument line shape for a Fourier Transform spectrometer is a cardinal sine or Sinc function. For interferograms that contain features that are at or near the resolution of the instrument (e.g. channel fringes), the natural instrument line shape can introduce artifacts in the calculated spectrum. These artifacts can be diminished by a technique referred to as apodization whereby the interferogram is multiplied by a tapering or apodizing function prior to transformation. A side effect of lowering the spectral artifacts is that the resolution of the resultant spectrum is also reduced.

In addition to some long-standing apodization functions, the Fourier transform package makes available a number of functions that optimize the tradeoff between reduction in the ringing artifacts and reduced resolution.

Implementation:
The apodization function is designed to accept a spectrometer detector interferogram (SDI) product as input. On a scan-by-scan and detector pixel-by-pixel basis, the interferograms in the input SDI product are multiplied by the chosen apodization function. The result is an SDI product that contains apodized interferograms.

Task: `RegSampledApodization`.
Mandatory Input Products: SDI.
Optional Input Products: None.
Output Products: SDI.