FTFitter_GUI

The FTFitter_GUI application provides a graphical user interface for the FTFitter general-purpose FTS spectral lineshape and continuum model fitting software. The application is written in IDL and is distributed as a precompiled .sav file which can be run in the IDL 8.4 virtual machine or full IDE environment.

FTFitter_GUI is designed as an interactive analysis tool to allow users to fit models to FTS spectra, in order to verify that the choice of model parameters is suitable, identify issues that prevent proper convergence of the minimization sequence, and produce line parameter templates that can be used in scripts calling FTFitter functions without user intervention for batch processing of spectra.

Version History

1.0	Aug 17 2011	Introduced.
1.1	Sept 1 2011	First release.
1.2	Sept 2 2011	Fixed resizing bugs and bugs in table colouring.
1.3	Sept 8 2011	Fixed multiple delete events bug.
1.4	Sept 13 2011	Fixed bugs with the parameter limits in the table.
1.5	Nov 18 2011	Fixed problem when reading FITS files of class SpectrometerPointSourceSpectrum with only one pixel.
1.6	Nov 21 2011	Added option to trim frequency range.
1.7	Dec 8 2011	Added integrated intensity calculation.
1.8		Support for SpectralSimpleCube SPIRE FITS files Support for IDL .sav files and reading in error spectrum
1.9		Compiled for IDL 8.2, additional SPIRE FITS file support
1.92	Sept 5 2013	Revised SPIRE FITS format support, FTS-2 FITS file support, bug fixes
1.93	Jan 13 2015	Compiled for IDL 8.4. FTFitter object revisions.

Contents

FTFitter Fundamentals

Fitting Algorithm

The general sequence of actions when using the GUI to perform a fit is as follows:

1. Load the spectrum.
2. Define a continuum model with suitable initial guesses for the parameters and fit to the data.
3. Select line positions or use the automatic peak detection tool to specify lines.
4. Perform a fit of the lines and continuum.
5. Repeat from 3. until all lines in the spectrum have been modeled, the residual is satisfactory, and no parameters are at their limits.

During each fitting sequence, the chi-squared residuals are minimized while varying all free parameters of the line and continuum models simultaneously. The minimization is performed using the Levenberg–Marquardt algorithm, which has the following advantages:

- Fast and relatively efficient compared to other minimization algorithms, provided there are fewer than ~1000 free parameters
- Parameter uncertainties are computed 'for free' from the covariance matrix
- Convergence is robust, even if initial guesses are off
- A well-established IDL library exists, written by Craig B. Markwardt of NASA/GSFC, based on the Fortran MINPACK-1 library
- The IDL implementation does not require analytical derivatives, which are not possible for the Fourier-space calculations performed in FTFitter

Simultaneous fitting of all line and continuum parameters is essential since many of the parameters are not orthogonal. For example, energy in the lineshapes affects the calculation of the continuum level, and sidelobes from neighbouring lineshapes can be superimposed. FTFitter accomplishes this efficiently by calculating the interferogram representing the superposition of either pure cosines (for unresolved lines) or the Fourier Transform of the theoretical lineshapes (for resolved lines), multiplying by the windowing function representing the instrumental lineshape (ILS), and then performing a single FFT to produce the model spectrum. This is more efficient that calculating the lineshapes individually and convolving with the ILS.

The following caveats must be observed:

- Accurate measurement errors must be provided in order for the parameter uncertainties to be meaningful.
- The fitting algorithm assumes that the model is a complete representation of the data; a model with too few or too many parameters will produce unrealistic uncertainty values.
- Large parameter uncertainties are generally a symptom of an incomplete model or low S/N, but small parameter uncertainties are not always a guarantee that the model is accurate.
- Parameter constraints (limits) may be added to improve the robustness of the fitting by preventing lines from 'wandering' or disappearing, but the uncertainties in the final parameters will not be valid if any parameters are pegged at the limits.

Lineshape functions

FTFitter simulates various lineshape models which, along with the continuum model(s), are effectively convolved with the instrumental lineshape (ILS) to produce the model spectrum. Each lineshape is defined by a maximum of 4 parameters, which are denoted here as *a*, *b*, *c*, and *d*, in the order they appear in the FTFitter GUI table.

- **Unresolved** (monochromatic)- The lineshape will be purely the ILS defined by the FT of the interferogram windowing function. Since the ILS width is fixed, the only relevant parameters for unresolved lines are the integrated area and the centre frequency.
- **Gaussian**- In terms of the FWHM, the Gaussian profile reduces to:
$$\varphi(\sigma) = a2^{\frac{-4}{c^2}(\sigma-b)^2}$$
where *a* is the amplitude, *b* is the centre frequency, and *c* is the FWHM.
- **Lorentzian**- In terms of FWHM, the Lorentzian profile becomes:
$$\varphi(\sigma) = \frac{ac}{2\pi\left((\sigma-b)^2 + \left(\frac{c}{2}\right)^2\right)}$$
where *a* is the amplitude, *b* is the centre frequency, and *c* is the FWHM.
- **Voigt**- The classical Voigt profile is computed in terms of damping parameter, *p*, and frequency offset, *u*, and scaled by the amplitude, *a*:
$$\varphi(\sigma) = a\frac{p}{\pi}\int_{-\infty}^{\infty}\frac{e^{-y^2}}{p^2+(u-y)^2}dy$$
where the frequency offset is defined in terms of the centre frequency, *b*, and Doppler width, *c*:
$$u = \frac{\sigma - b}{c}$$
and the damping parameter is defined in terms of the transition rate, *d*, and Doppler width, *c*:
$$p = \frac{d}{4\pi c}$$

Continuum Models

The spectral baseline or continuum can be modeled using the following functions

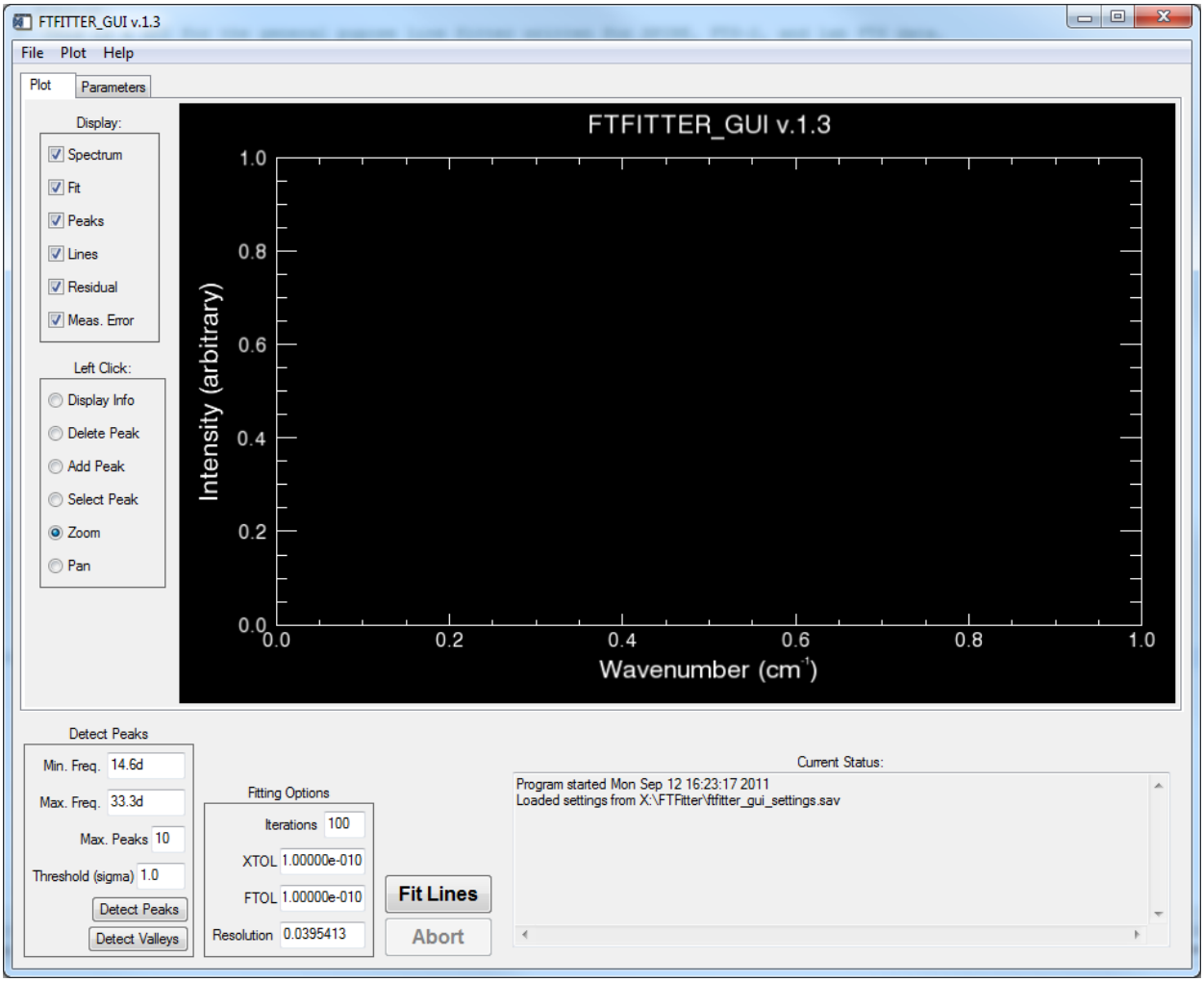
- **Planck Greybody**- A blackbody function of given temperature, *b*, scaled by an emissivity, *a,* with frequency, *σ*, in units of cm^{-1}:
$$\varphi(\sigma) = a\frac{2hc^2\sigma^3 * 10^4}{(e^{\frac{100hc\sigma}{kb}} - 1)}$$
note that in this case *c* is the speed of light (in m/s) and not the parameter c.
- **Polynomial**- A third order polynomial baseline of the following form:
$$\varphi(\sigma) = a + b\sigma + c\sigma^2 + d\sigma^3$$
- **Poly2**- A higher order polynomial to be used with the cubic polynomial baseline:
$$\varphi(\sigma) = a\sigma^4 + b\sigma^5 + c\sigma^6 + d\sigma^7$$

Launching the FTFITTER_GUI application

If using a full version of IDL (8.1 or later), you can include the installation folder in your IDL search path or manually load the code by typing: *restore,'pathname'* , where *pathname* is the fully qualified path to **ftfitter_gui.sav**. Typing *ftfitter_gui* in the command line will launch the GUI. If you are using the IDL Virtual Machine (VM), then you must select the ftfitter_gui.sav file when starting IDL.

When starting, the GUI searches for a settings file named **ftfitter_gui_settings.sav** in the current directory. If this file is not found, default settings will be used -so be careful to set reasonable values for all parameters before attempting a fit. Upon exit, the GUI will store the current program parameters to the **ftfitter_gui_settings.sav** file.

The main screen of the FTFitter GUI is shown below:



There is a **Plot** tab for displaying the current lines, spectrum, and model, and a **Parameters** tab which provides a table for viewing and editing individual line parameters.

Plot Tab

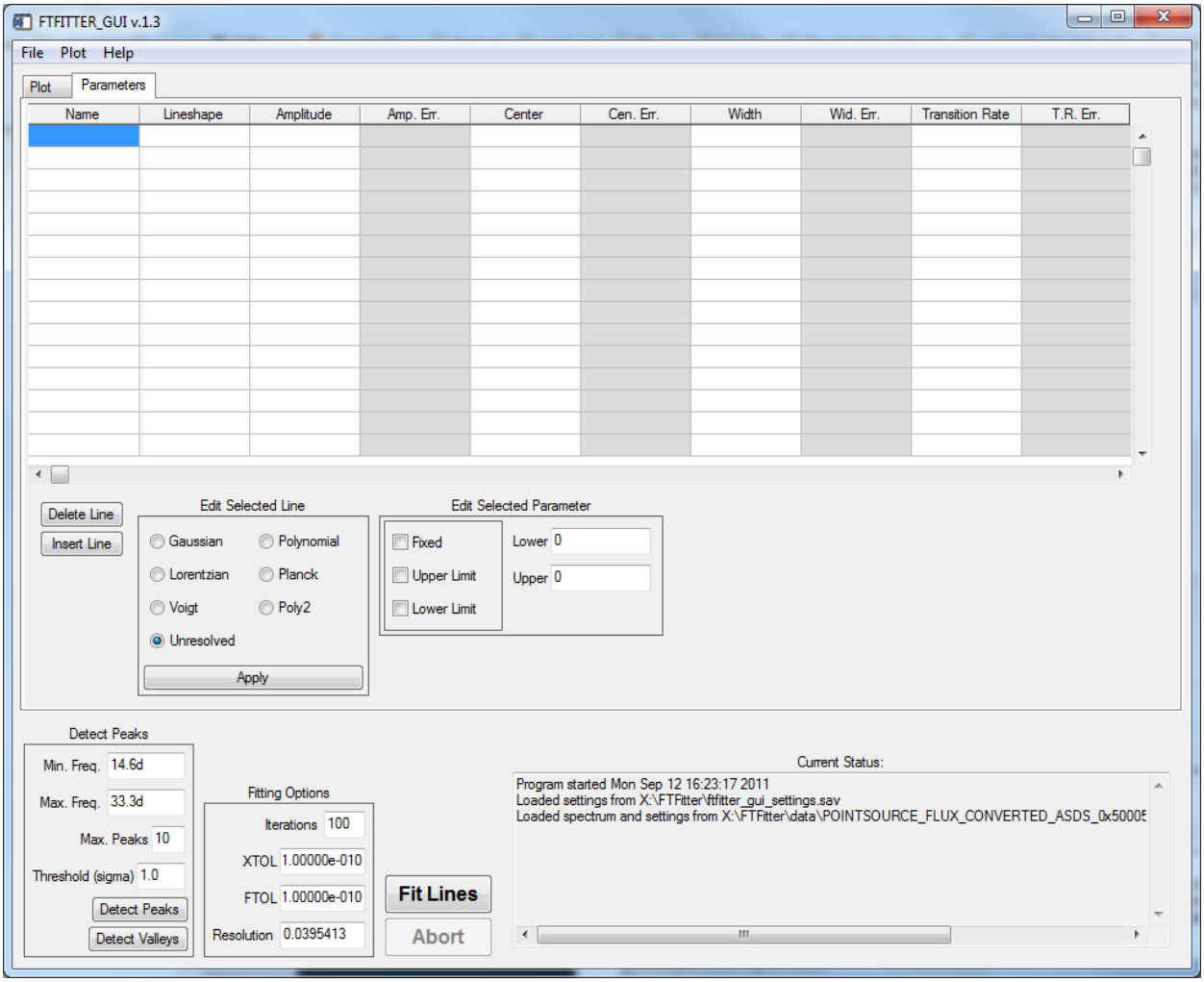
The plot tab includes an interactive plot on the right, and a column of checkboxes and radio buttons on the left. The plot can be manipulated using the mouse buttons. By default, the left mouse button zooms in the frequency scale; the right button restores the previous zoom value; and the middle mouse button autoscales the amplitude scale to the data. If available, the scroll wheel incrementally zooms in on the current mouse location. The displayed values and the actions associated with the mouse buttons are controlled by the buttons on the left:

- The '**Display**' checkboxes determine which features are shown in the interactive plot:
    - '**Spectrum**' is the loaded spectral data.
    - '**Fit**' is the current model spectrum.
    - '**Peaks**' are the frequency positions of the lineshapes.
    - '**Lines**' are the individual lineshape models.
    - '**Residual**' is the difference between the spectrum and the current fit.
    - '**Meas. Error**' is the loaded measurement error used for the weighting function.
- The '**Left Click**' radio buttons determine what occurs when the left mouse button is used in the plot window:
    - '**Display Info**' provides a dynamic display of the currently selected trace name and value.
    - '**Delete Peak**' allows the user to delete peaks by selecting them with the mouse instead of using the parameters table.
    - '**Add Peak**' creates a new default lineshape at the selected peak position.
    - '**Select Peak**' highlights the clicked peak in the parameter table.
    - '**Zoom**' allows a zoom window to be used while clicking and dragging the mouse. Right clicking restores the previous zoom. The scroll wheel can also be used to zoom in or out on the current mouse location. The original zoom can be restored by using the '**Plot | Autoscale X Axis**' and '**Plot | Autoscale X Axis**' menu items.
    - '**Pan**' allows the plot to be panned to a new region without affecting the scaling.

Parameters Tab

The Parameters tab (shown below) provides a table to enter, view and edit continuum and lineshape parameters, as well as fields to set limits on selected parameters.

The table (shown below) lists the currently defined continuum and line parameters. Values are updated after the fit, and can be edited by the user.



- Cells are colour-coded with the following scheme:
    - Dark Blue – the currently selected cell
    - White - editable cell
    - Grey - non-editable cell (e.g. the uncertainty values)
    - Pink - parameter has at least one limit defined
    - Red - parameter value is pegged at a limit
    - Light Blue - the parameter is fixed
    - Yellow - the parameter uncertainty is suspiciously large
- Multiple cells can be selected at once to apply changes in parallel.

- The column labels represent the meanings of the parameters for the currently selected line. Note that columns that are not relevant to the currently selected lineshape are labeled with '-'. Any values in these cells will not be considered in calculating those lineshapes.
- The lines may be sorted by selecting the label of the column over which to sort.

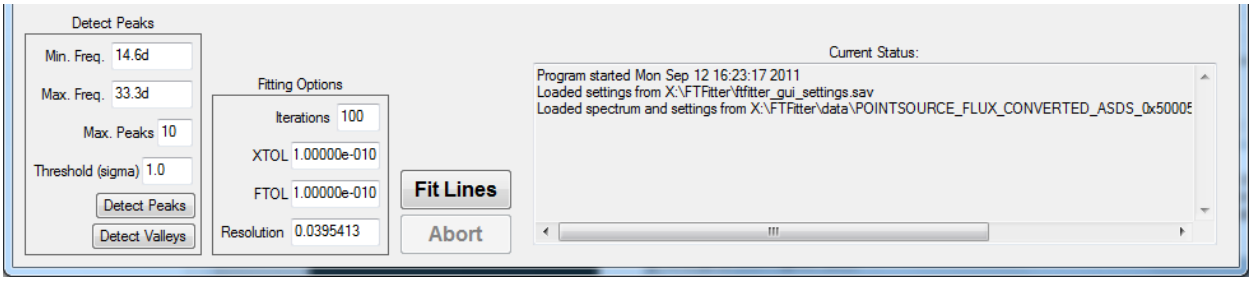
Lines may be created by selecting the '**Insert Line**' button. A new line with default values is inserted at the currently selected location in the list. The '**Delete Line**' button deletes the currently selected line(s). Note that there is no undo feature.

The '**Edit Selected Line**' box allows the selected line(s) to be changed to the currently selected lineshape or continuum model. When the '**Apply**' button is clicked, the change(s) will take effect.

The '**Edit Selected Parameter**' box allows the limits for the currently selected parameter(s) to be changed. The '**Fixed**' checkbox determines if the parameter is held constant during the optimization. The '**Upper Limit**' and '**Lower Limit**' checkboxes determine if the upper and lower limits defined on the right take effect on the selected parameter(s). Note that some parameters have mandatory limits. For example, the lineshape centre frequency is limited to the data range, and the line widths may not be negative.

Fitting Options

Once a spectrum has been loaded and one or more continuum models or lineshapes have been defined, the optimization sequence can be controlled by the buttons on the bottom of the screen.



- The fields in the '**Detect Peaks**' box set the options for the automatic peak detection algorithm. When the '**Detect Peaks**' button is pressed, the algorithm looks for peaks in the residual spectrum in the frequency range defined by '**Min. Freq.**' and '**Max. Freq.**', with amplitude higher than the specified number of standard deviations away the mean, up to a maximum number of peaks defined by '**Max. Peaks**'. It is usually best to detect small numbers of peaks at a time, and then perform a fit before detecting more peaks. The '**Detect Valleys**' button is used to detect local minima (absorption lines) instead of local maxima (emission lines). By default, peaks are defined as unresolved lineshapes, since there is no heuristic for automatically detecting the lineshape.
- The '**Fitting Options**' fields define the maximum number of iterations and the tolerances for the convergence criteria:
    - '**Iterations**' defines the maximum number of iterations to perform. It is usually best to keep this number small. If the maximum number of iterations is exceeded, the fit will not have converged and the parameters should not be trusted.
    - '**XTOL**' measures the relative error desired in the approximate solution. The default is 1e-10.
    - '**FTOL**' measures the relative error desired in the sum of squares. The default is 1e-10.
    - '**Resolution**' defines the linewidth of the unresolved lineshapes, and should be matched to the true resolution of the spectrum, not including any zeropadding. The resolution is not varied during the minimization sequence.
- The '**Fit Lines**' button begins a minimization sequence using the currently defined lineshapes and convergence parameters.
- The '**Abort**' button stops a minimization sequence that is in progress.

Appendix A: Example Fit Sequence

A typical fitting sequence is outlined below.

1. Load spectrum
    - Open an existing FITS or SPC file with the data to fit.
2. Add continuum
    - Define the continuum\baseline parameters by inserting a line on the Parameters tab.
3. Fit the continuum
    - The continuum should be modeled before an attempt is made at detecting or fitting lines.
4. Add lines or use the peak detection feature
    - Lines can be added\removed using the Plot tab or the Parameters tab. Adjust the initial parameter values if necessary in the table.
5. Set limits
    - Sensible limits should be set on the centre frequencies and any other parameters that can be constrained by a priori knowledge. The amplitude of absorption lines should be limited to negative values; emission lines should be limited to positive values.
6. Set the convergence tolerance and maximum iterations
7. Perform the fit
8. Adjust parameters and repeat fit if necessary
    - If some lines have wandered off the peak during fitting, or closely spaced lines are interacting to produce non-physical results, the parameters and\or limits should be adjusted manually and the fit should be repeated.
9. Save\export lines
    - Once the fit is satisfactory, the model spectrum and parameters can be saved. When saved in XML format, the line parameters can be loaded in subsequent runs or used in custom IDL scripts to batch process multiple similar spectra.

Appendix B: Instrumental Lineshapes

When measured by an FTS, the spectrum of a monochromatic source is distorted by the following effects:

- the finite length of the Fourier transform (i.e. the windowing of the interferogram)
- apodization due to vignetting and optical spillover
- optical misalignment
- diffraction or dispersion within the instrument
- the divergence or obliquity effect, resulting from differing optical path lengths for off-axis rays
- phase distortion due to optical or electrical effects

This distorted spectrum is referred to as the instrumental lineshape (ILS). For a well-designed FTS, the interferogram windowing function is the dominant contributor, and the ILS is commonly approximated by the Fourier transform of an ideal rectangular windowing function, which is the Sinc function. The monochromatic line is then represented by a Sinc of width, w, (determined by the length of the interferogram) and line frequency, σ_c,:

$$ILS = \frac{\sin\left(\frac{2\pi}{w}(\sigma - \sigma_c)\right)}{\frac{2\pi}{w}(\sigma - \sigma_c)}$$

This approximation is only valid for the continuous Fourier transform of the source. For the discrete Fourier transform, if the interferogram is not strictly band-limited, frequency components above the Nyquist frequency will be aliased into the spectrum. Furthermore, the Fourier transform contains both positive and negative frequencies. Since the oscillations of the Sinc function extend infinitely (at diminishing amplitude), the tail of the negative copy of the ILS will extend into the positive domain. This effect is larger for lower frequencies, but a more correct approximation of the ILS for an ideal FTS is the sum of the positive and negative Sinc functions:

$$ILS = \frac{\sin\left(\frac{2\pi}{w}(\sigma - \sigma_c)\right)}{\frac{2\pi}{w}(\sigma - \sigma_c)} + \frac{\sin\left(\frac{2\pi}{w}(\sigma_c - \sigma)\right)}{\frac{2\pi}{w}(\sigma_c - \sigma)}$$

For SPIRE data, the contribution of the negative Sinc ripples is on the order of a few percent of the line amplitude. If the spectra are fit using a simple Sinc function, there will be a sinusoidal ringing in the residual at this level, which will either skew the fit line parameters, or masquerade as weak lines after strong lines have been subtracted. The figure below shows a comparison of

For broadened lines, the spectrum will be the convolution of the source spectrum and the ILS. As the lines get broader, the effect of the negative frequency ringing is diminished.

Instead of fitting a theoretical ILS function directly to the spectrum, FTFitter simulates an interferogram of the source spectrum using either monochromatic or broadened lineshapes, applies a windowing function that represents the instrument apodization and finite sampling, and then performs a FFT to produce the model spectrum. In this way, the negative frequency components of monochromatic lines are accounted for and broadened lines are simulated without requiring convolution in the spectral domain.

It must be noted that while FTFitter can properly simulate the effects of the DFT sampling and the apodization due to a real-valued windowing function, other effects such as phase or off-axis divergence in the FOV of a single pixel must be corrected before fitting.

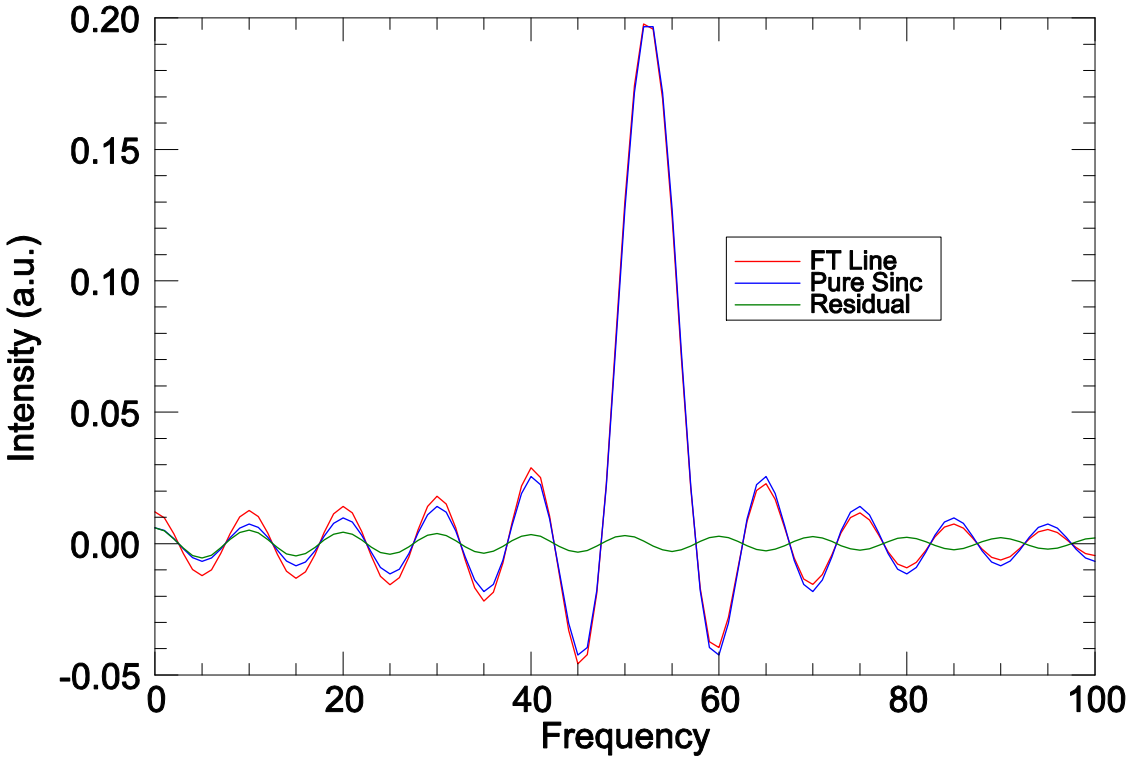


Figure 1. Pure Sinc function compared to the DFT of a monochromatic line.

Appendix C: Error Analysis

While FTFitter generally succeeds in finding the global minimum in the chi-squared parameter space, care must be taken in the interpretation of the parameter uncertainty estimates. In general, the uncertainty estimates for a well-defined model will be proportional to the S/N in the spectrum. An attempt was made to characterize the relationship between S/N and fit uncertainties by simulating a spectrum consisting of a pair of known line profiles with a given separation and superimposing a randomly generated white noise spectrum, then performing a fit using a given initial guess, over a large number of individual trials.

Spectra were simulated with pairs of lines of FWHM of 0.08 separated by 0.16, 0.12, and 0.08, with a spectral resolution of 0.04. White noise was added to produce spectra with S/N of 100, 10, and 2. A sample of the simulated spectra for one of the 1000 trials is shown below.

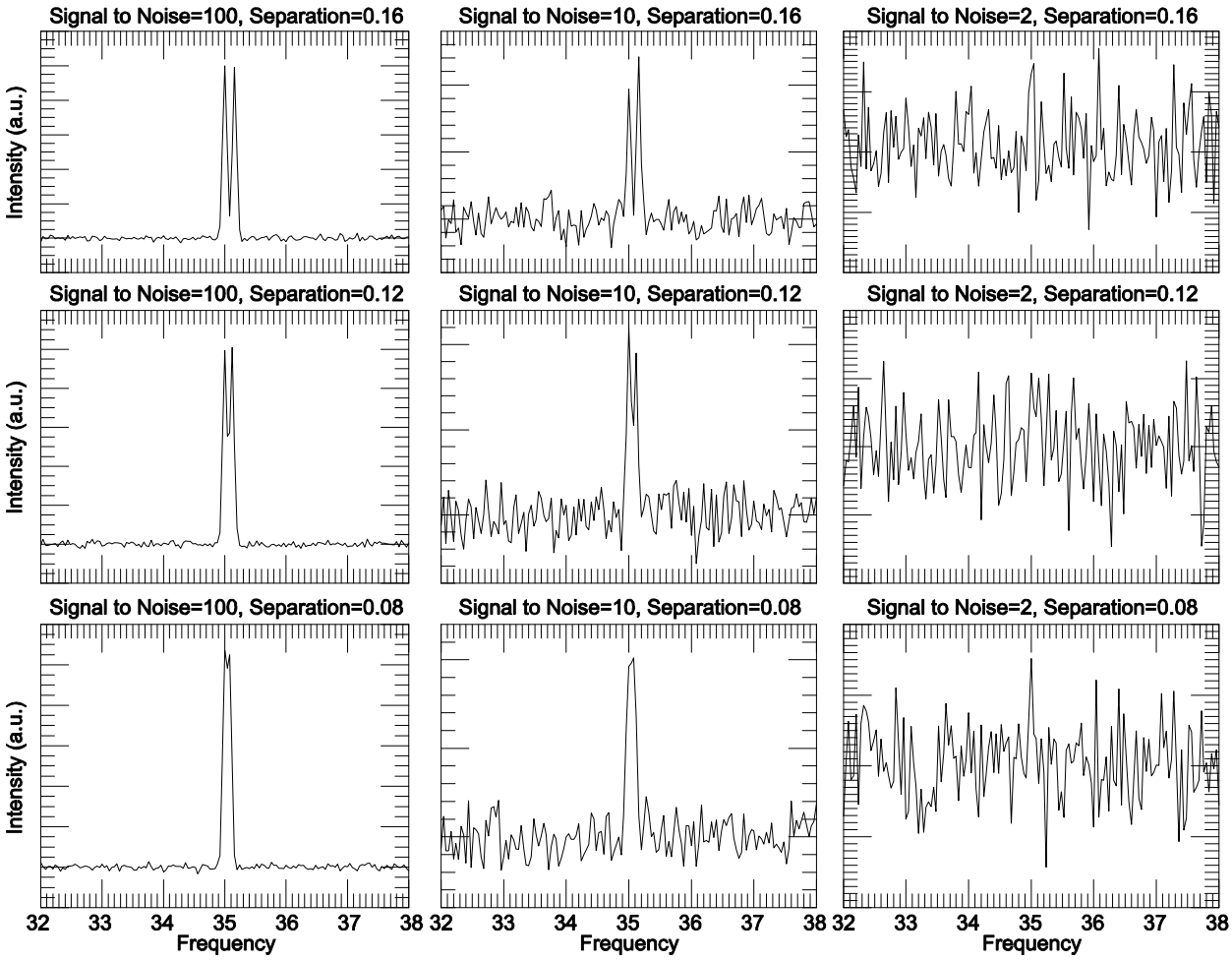


Figure 2. Simulated spectral line pairs at S/N levels of 100, 10 and 2.

Spectra from each noise trial were fit and the variations of the returned parameters were compared. Figure 3 shows histograms of the returned line centre as a function of line spacing and S/N level. As expected, the variation in the line centre for S/N of 2 is quite large. The fitting appears to still be able to identify the line separation in the low S/N scenario since the initial guess provided was close to the true values.

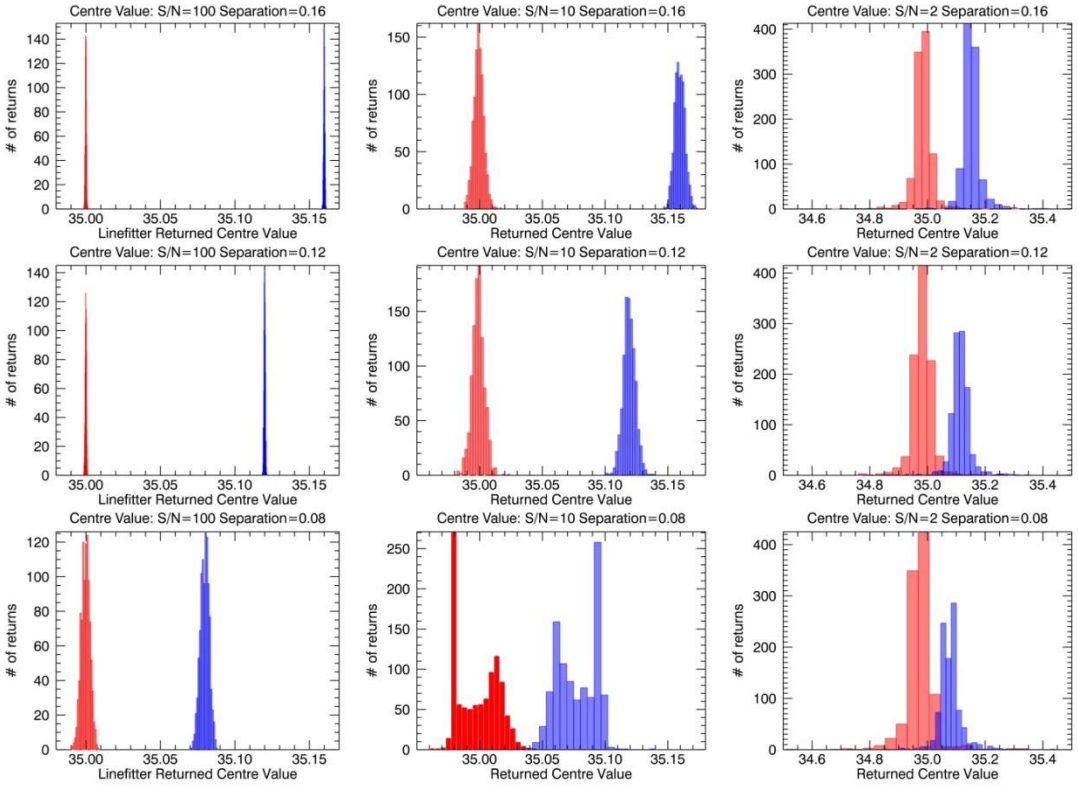


Figure 3. Histograms of the returned line centre parameter values for different line separation and spectral S/N levels.

The important question is whether the standard deviation of the returned parameters over a large number of random trials is well characterized by the parameter uncertainty returned in a single fit. The ratio of the standard deviation of the returned parameter value to the mean uncertainty over the 1000 trials is shown in Figure 4. The parameter uncertainty matches the standard deviation of the parameter values for high S/N and large line separation. As the S/N falls below ~60 for closely spaced lines or below ~10 for widely spaced lines, the average returned uncertainty increases drastically. The distribution of returned uncertainty values is not Gaussian for low S/N values; there are an increasing number of outliers with extremely large uncertainty values as the S/N decreases, which causes the mean uncertainty to increase although the mode is still near the expected value.

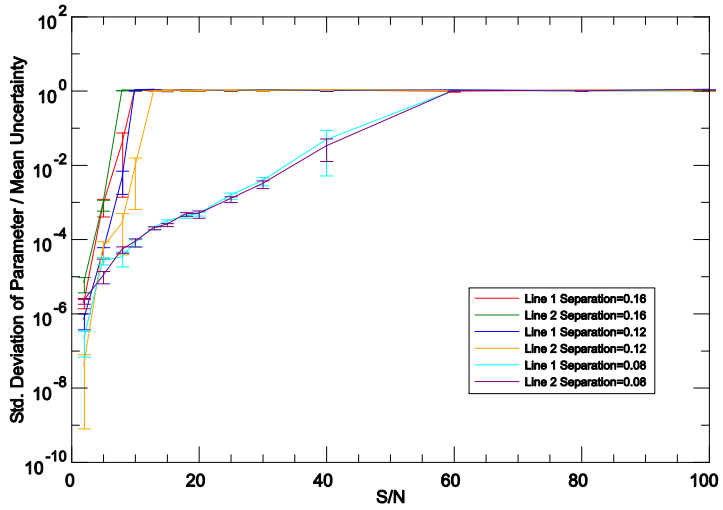


Figure 4. Ratio of the standard deviation of the returned parameter to the mean parameter uncertainty over 1000 trials.

The occurrence of unrealistically large uncertainty estimates does not seem to be deterministic. Often, high uncertainty values correspond with instances in which the number of minimization iterations is larger than normal, but this is not always the case, as can be seen in Figure 5 and Figure 6. For a S/N of 5, there is seemingly a direct correspondence between high uncertainty values and larger than normal number of iterations, and it also appears that the high uncertainty values correspond with cases where the line centre converged on one side or the other of the true centre. For a S/N of 2, however, it can be seen that in about 20% of the cases, the returned uncertainties are not realistic, and there is no clear trend between the uncertainty value and the number of iterations, residual, or returned line centre value. The same behaviour is observed in the other line parameters.

To a certain extent, the uncertainties of the amplitude, width and centre are related, and approximately the same fraction of trials result in anomalously large uncertainty values for each of these parameters.

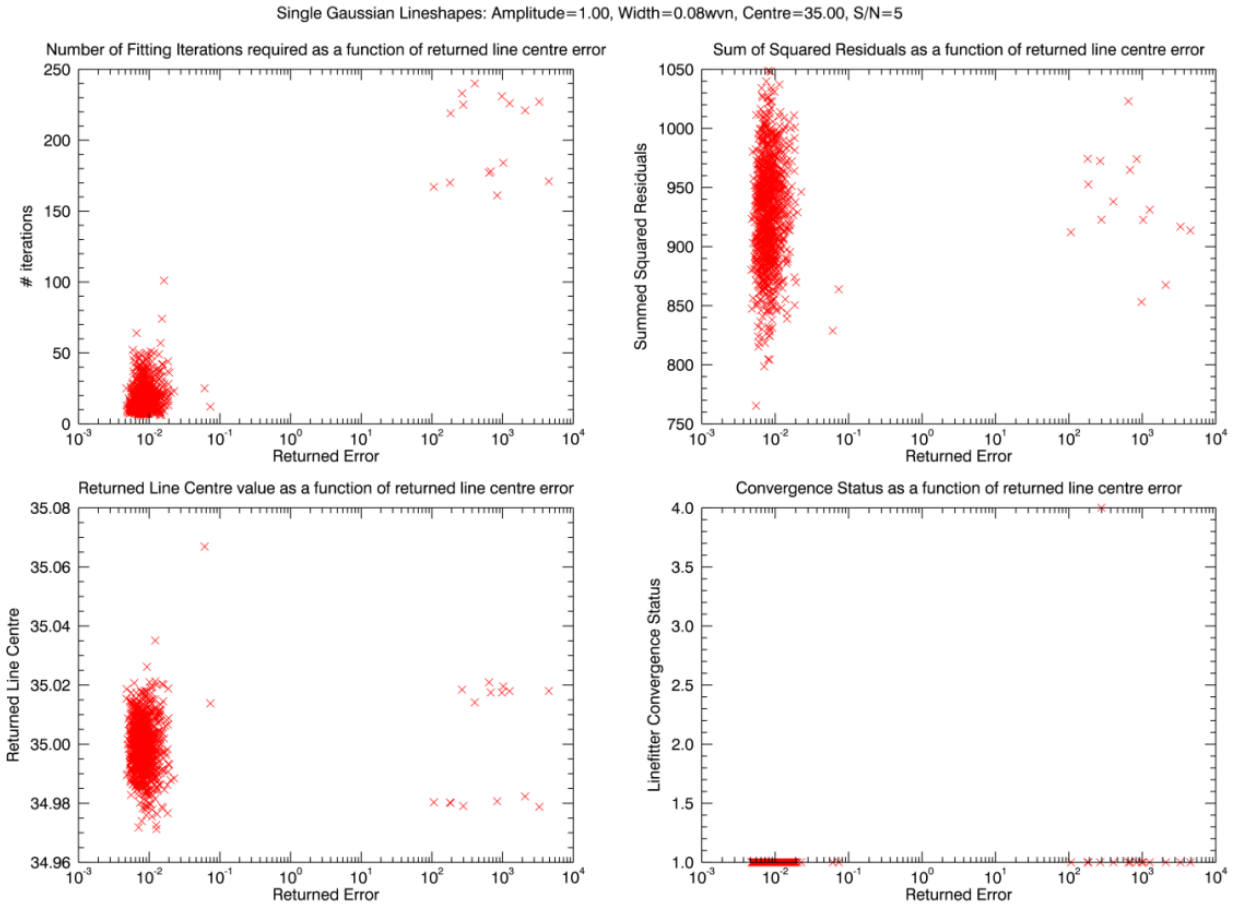


Figure 5. Trends between the returned uncertainty value and: Number of iterations, residual, line centre value and convergence status, for a S/N of 5.

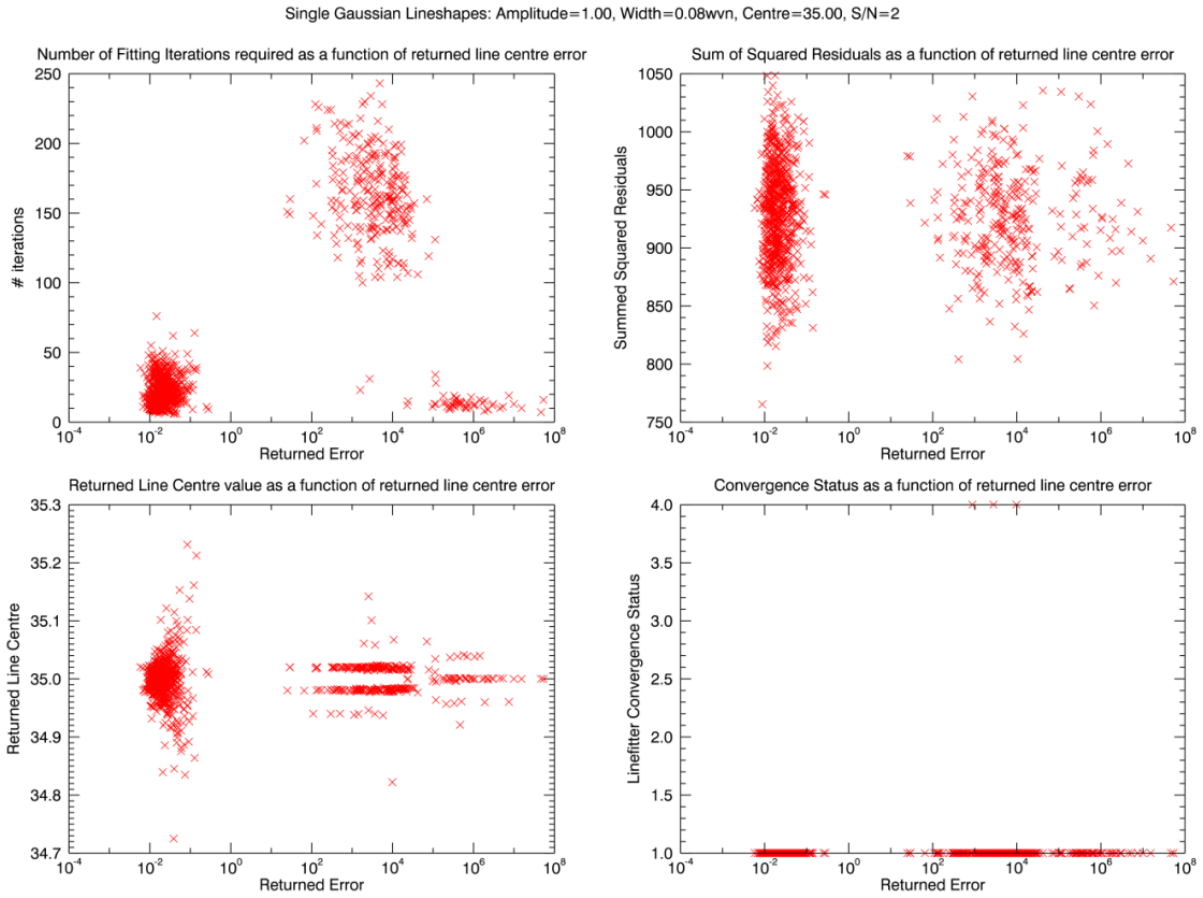


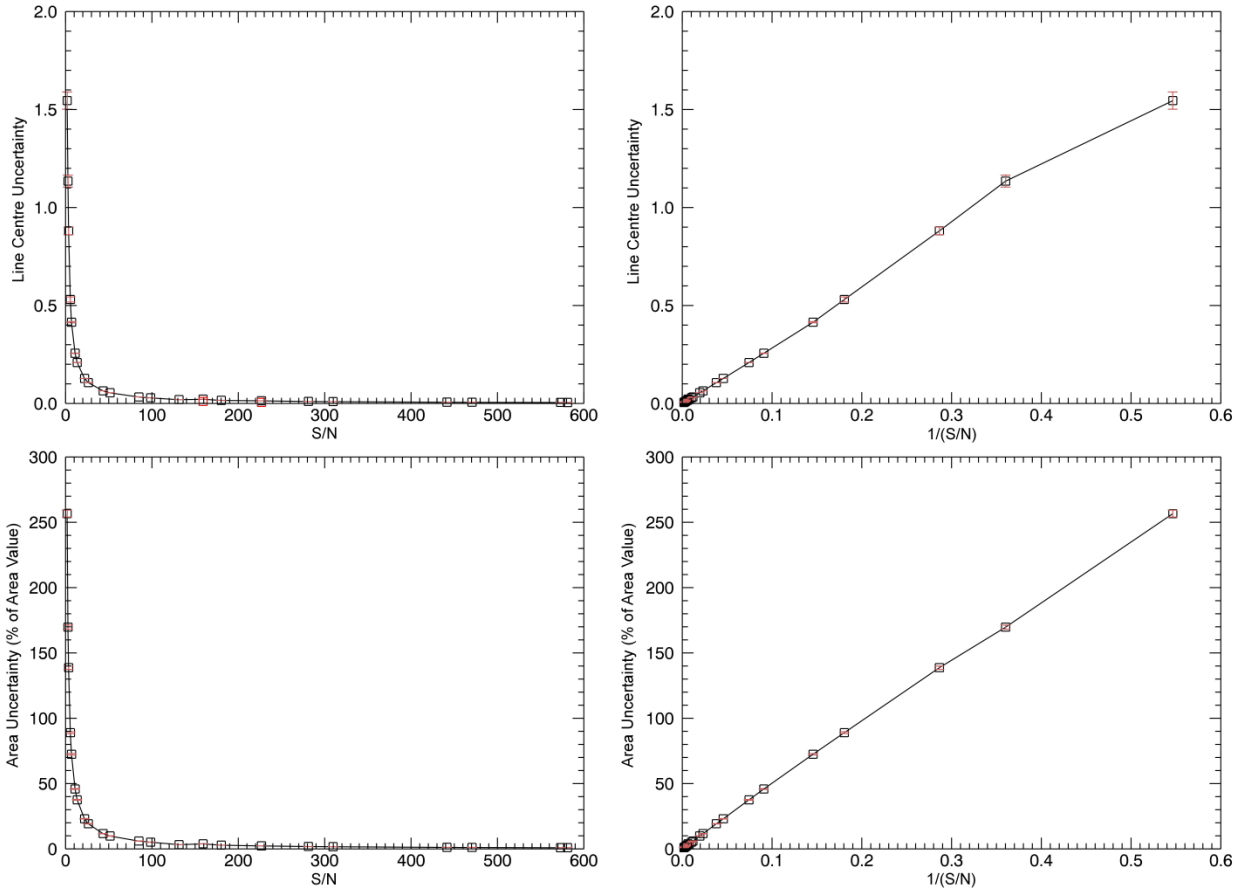
Figure 6. Trends between the returned uncertainty value and: Number of iterations, residual, line centre value and convergence status, for a S/N of 2. There relationship between the parameters and uncertainty is ambiguous.

There is no obvious correlation between the anomalously high uncertainty values and any other metric of the fit quality. Consequently, users should be aware that fitting at low S/N is problematic, and the occurrence of parameter uncertainties which are much larger than the parameter value is an indication that the fit has not converged properly. Furthermore, the following criteria must be met in order for the uncertainty estimates to be valid at all:

- The measurement errors must be Gaussian in nature
- There must be a linear relationship between a change in a parameter and a change in a data value, at least for small changes
- The model must be an accurate and complete description of the measured data
- The measurement errors being used to weight the data must be valid
- There must be no limits on the parameters, or at least the limits must be far from the optimum parameter values.

Ignoring cases where the uncertainty estimates are obviously too large, we can derive a rule of thumb for the relationship between S/N and expected line centre uncertainty, for typical emission line spectra. A spectrum of C, CO, and ^{13}CO lines with a range of amplitudes was simulated, and the noise level was varied to produce S/N ranging between 1.5 and 500. For each S/N level, 1000 fitting trials were performed using different Gaussian noise seeds. The average line centre uncertainty and line area uncertainty as a function of the S/N are shown below. For S/N > 2, the trend is linear with the reciprocal of

S/N, as expected. The plots show the line centre uncertainty represented as a fraction of a resolution element, and the area uncertainty as a percentage of the area value.



Appendix D: Integrated Intensities

The integrated intensity for a given lineshape is the integrated area under the line profile. The uncertainty on the integrated intensity is calculated from the error propagation of the uncertainties in the parameters, assuming the uncertainties are accurate representations of the standard deviations of the parameter values. This uncertainty should be considered an upper limit in the case of the resolved lineshapes, as the code currently does not take into account the Sinc convolution which takes place during the minimization. FTFitter computes the uncertainty in the integrated intensities as follows.

$$Area = \int_{\sigma_{lo}}^{\sigma_{hi}} f(\sigma) d\sigma$$

$$\Delta Area = \sqrt{\left(\Delta a \frac{\partial Area}{\partial a}\right)^2 + \left(\Delta b \frac{\partial Area}{\partial b}\right)^2 + \cdots}$$

- **Unresolved** (monochromatic)- There are only two parameters for the unresolved lineshape: the centre frequency and the integrated intensity. The uncertainty in the integrated intensity is taken directly from the fitting results.
- **Gaussian**- The area under the Gaussian profile is given by:

  $$Area = \frac{\sqrt{\pi}}{2\sqrt{ln2}} ac$$

  Where *a* is the amplitude and *c* is the FWHM. The uncertainty in the area for an amplitude uncertainty of Δa and a width uncertainty of Δc is given by:

  $$\Delta Area = \frac{\sqrt{\pi}}{2\sqrt{ln2}} \sqrt{(a\Delta c)^2 + (c\Delta a)^2}$$

- **Lorentzian**- The Lorentzian profile is normalized, so the integrated intensity is equal to the scale factor parameter. The uncertainty is taken directly from the parameter uncertainty.
- **Voigt**- The Voigt profile is normalized, so the integrated intensity is equal to the scale factor parameter. The uncertainty is taken directly from the parameter uncertainty.
- **Polynomial-** The area under the polynomial baseline is computed as:

  $$\begin{aligned} Area &= \int_{\sigma_{lo}}^{\sigma_{hi}} a + b\sigma + c\sigma^2 + d\sigma^3 \, \mathrm{d}\sigma \\ &= a(\sigma_{hi} - \sigma_{lo}) + \frac{b}{2}({\sigma_{hi}}^2 - {\sigma_{lo}}^2) + \frac{c}{3}({\sigma_{hi}}^3 - {\sigma_{lo}}^3) + \frac{d}{4}({\sigma_{hi}}^4 - {\sigma_{lo}}^4) \end{aligned}$$

  The uncertainty in the area is given by:

  $$\Delta Area = \sqrt{\left(\Delta a(\sigma_{hi} - \sigma_{lo})\right)^2 + \left(\frac{\Delta b}{2}({\sigma_{hi}}^2 - {\sigma_{lo}}^2)\right)^2 + \left(\frac{\Delta c}{3}({\sigma_{hi}}^3 - {\sigma_{lo}}^3)\right)^2 + \left(\frac{\Delta d}{4}({\sigma_{hi}}^4 - {\sigma_{lo}}^4)\right)^2}$$

- **Poly2-** The area under the higher order polynomial baseline is computed as:

  $$\begin{aligned} Area &= \int_{\sigma_{lo}}^{\sigma_{hi}} a\sigma^4 + b\sigma^5 + c\sigma^6 + d\sigma^7 \, \mathrm{d}\sigma \\ &= \frac{a}{5}({\sigma_{hi}}^5 - {\sigma_{lo}}^5) + \frac{b}{6}({\sigma_{hi}}^6 - {\sigma_{lo}}^6) + \frac{c}{7}({\sigma_{hi}}^7 - {\sigma_{lo}}^7) + \frac{d}{8}({\sigma_{hi}}^8 - {\sigma_{lo}}^8) \end{aligned}$$

  The uncertainty in the area is given by:

$$\Delta Area = \sqrt{\left(\frac{\Delta a}{5}(\sigma_{hi}{}^{5} - \sigma_{lo}{}^{5})\right)^2 + \left(\frac{\Delta b}{6}(\sigma_{hi}{}^{6} - \sigma_{lo}{}^{6})\right)^2 + \left(\frac{\Delta c}{7}(\sigma_{hi}{}^{7} - \sigma_{lo}{}^{7})\right)^2 + \left(\frac{\Delta d}{8}(\sigma_{hi}{}^{8} - \sigma_{lo}{}^{8})\right)^2}$$

- **Planck Continuum-** The Planck greybody continuum is integrated numerically over the spectral range. The uncertainty in the integrated area is estimated based on the parameter errors and the numerical partial derivatives of the integrated area with respect to the parameters. Note that fitting the Planck function is problematic when the peak of the blackbody curve is not in the spectral range, and the calculated uncertainties may not be realistic.