Last Updated: 03-03-2018

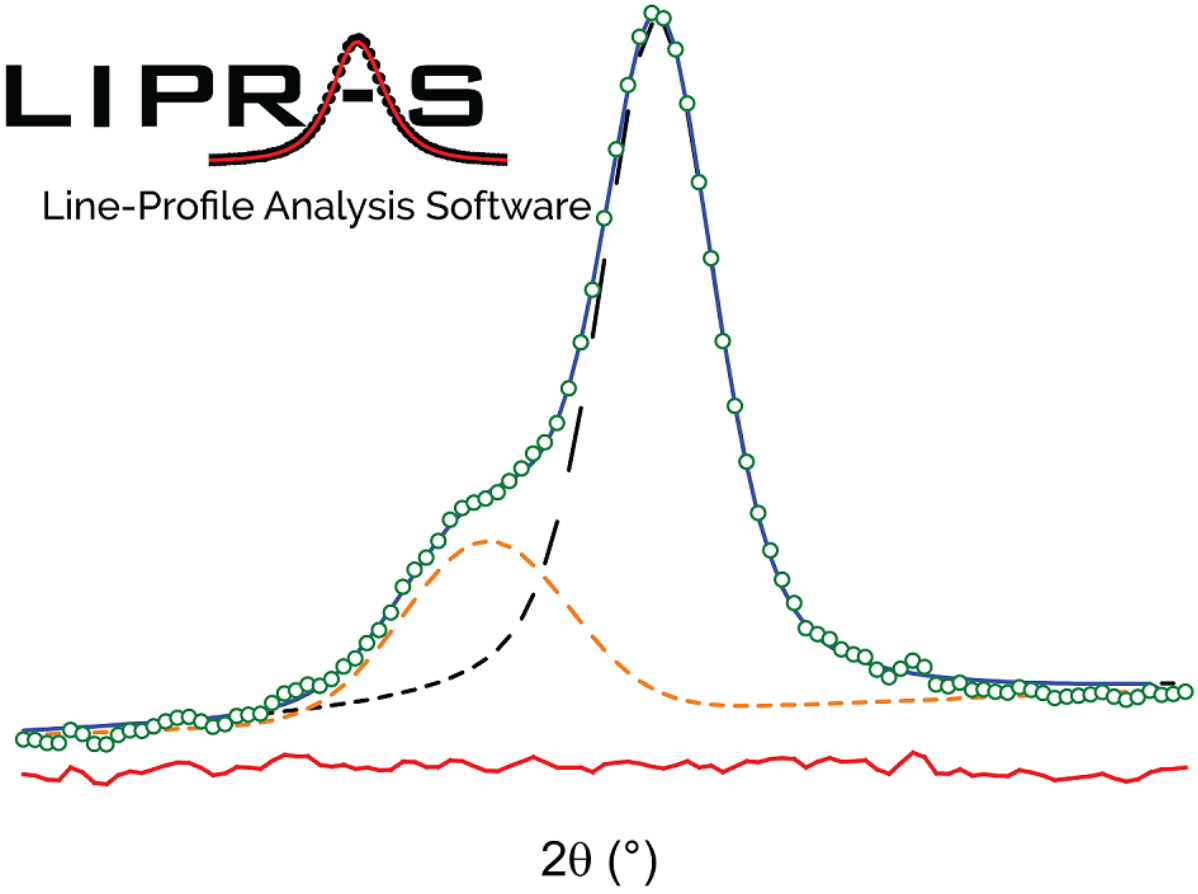


TABLE OF CONTENTS

History

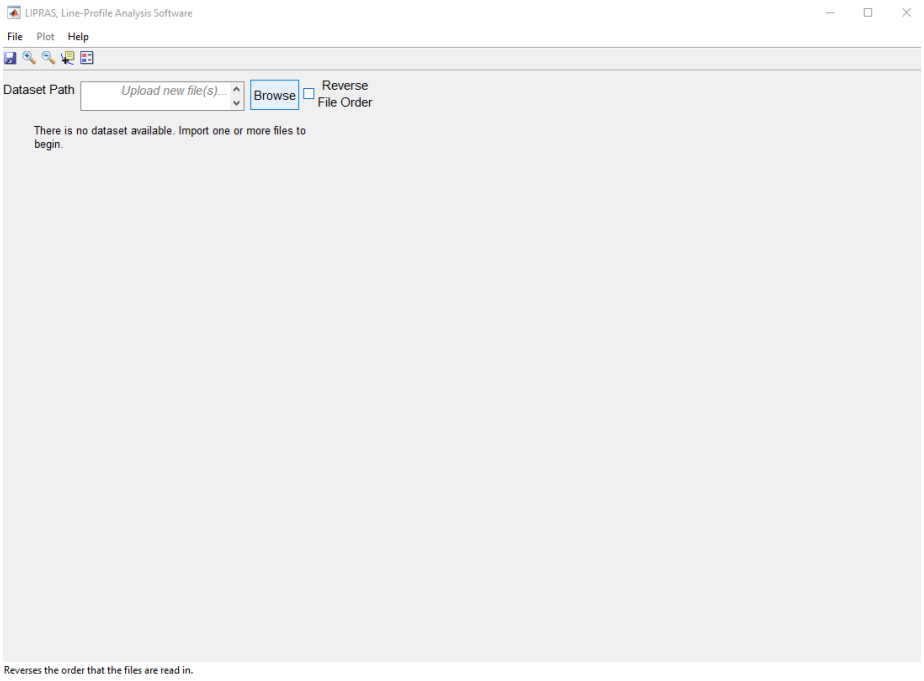
The origin of LIPRAS started back in 2006 when Jacob Jones was a post-doctoral researcher. Jacob created the backbone of the code responsible for subtracting a background and performing single peak fitting. In 2014, Chris M. Fancher continued development of the code created by Jacob and successfully turned it into a class file within MATLAB, and allowed for more user customizability and easier implementation of features. Giovanni Esteves further built upon Chris's version by polishing the code and adding features to enhances its feasibility to new users. Finally, in 2016 Klarissa Ramos joined the Jones Research group and jointly with Giovanni they refactored the class file and created a graphical user interface (GUI) which is now known as LIPRAS.

Peak Fitting with LIPRAS

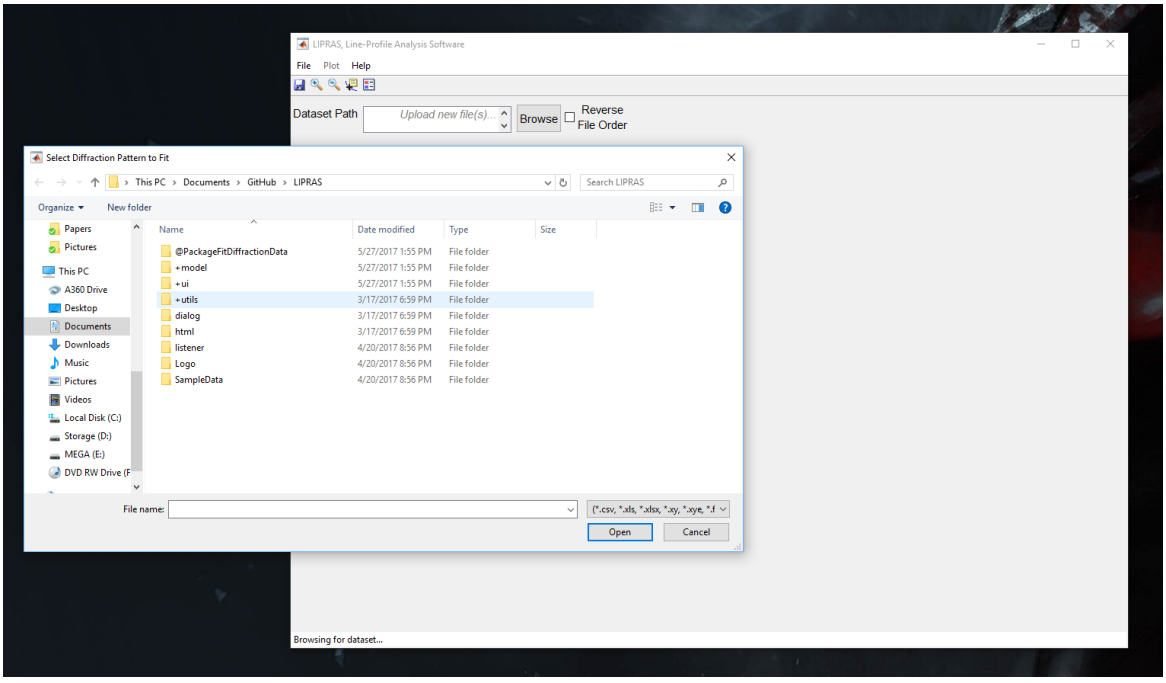
1. Browsing for Data and Accepted Data Files

LIPRAS except data in 2-3 column format. For files of type FXYE, CHI, and XRDML LIPRAS considers these special exceptions since they are generated from known sources that use a consistent file format. The XRDML for example is from Panalytical Laboratory diffractometers, this file type automatically reads in scans and sets the Cu-Kα1 and Kα2 wavelengths automatically upon reading the file. CHI files are from Fit2D which is a well-known program to reduce 2D diffraction images into 1D intensity vs. 2θ diffraction patterns. This file type skips the first 3 lines and begins to read the first 2 columns of data. Lastly, FXYE files come from beamline 11-BM-B of the Advanced Photon Source at Argonne National Laboratory. These files have 3 columns of 2θ, intensity, and error. The file formats XY and XYE cannot have any headers. The XY file type reads in two column data of 2θ and intensity while the XYE reads in 3 columns of 2θ, intensity, and error. The last file types, XLSX, XLS, and CSV (Windows OS only) are Excel files. These can have some header to them but it is recommended that it be limited to 4 lines or less. In any case, the best scenario is if these files do not have any header at all and are just 2 or 3 column formats of X,Y or X,Y,Error.

Click the 'Browse' button to begin



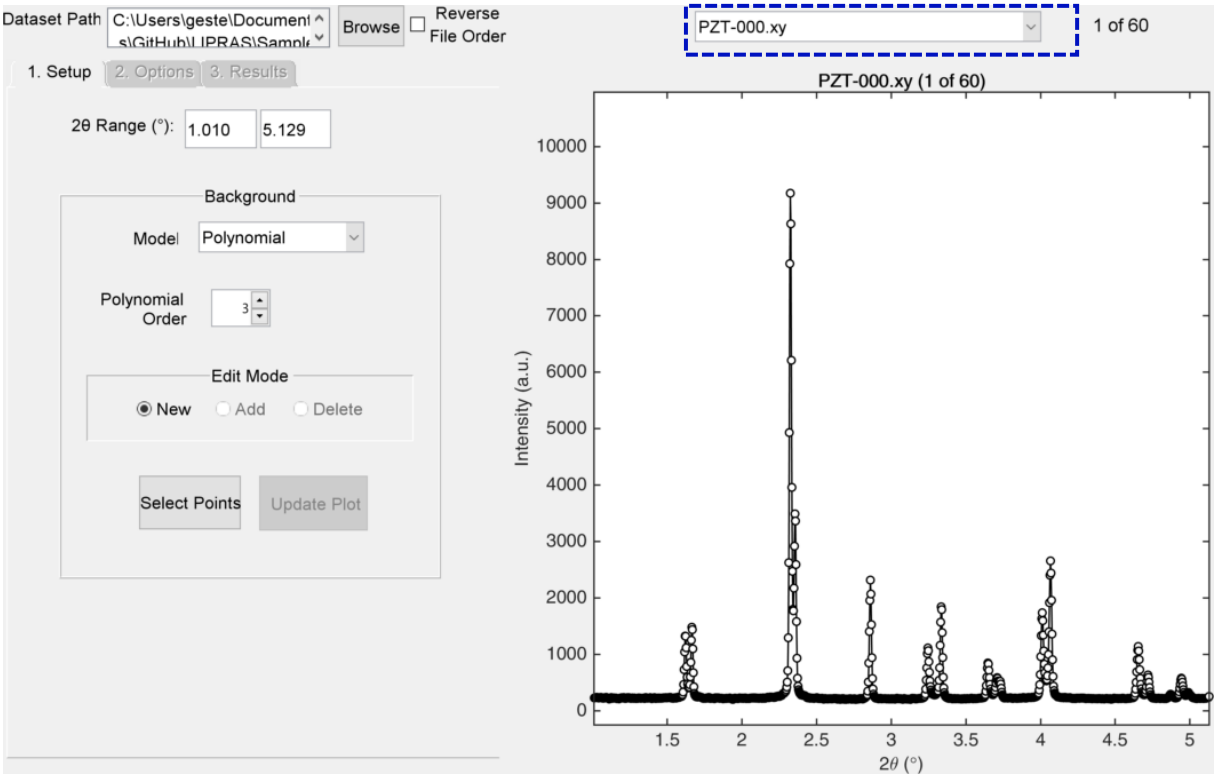
The File Explorer will prompt the user to select the data files



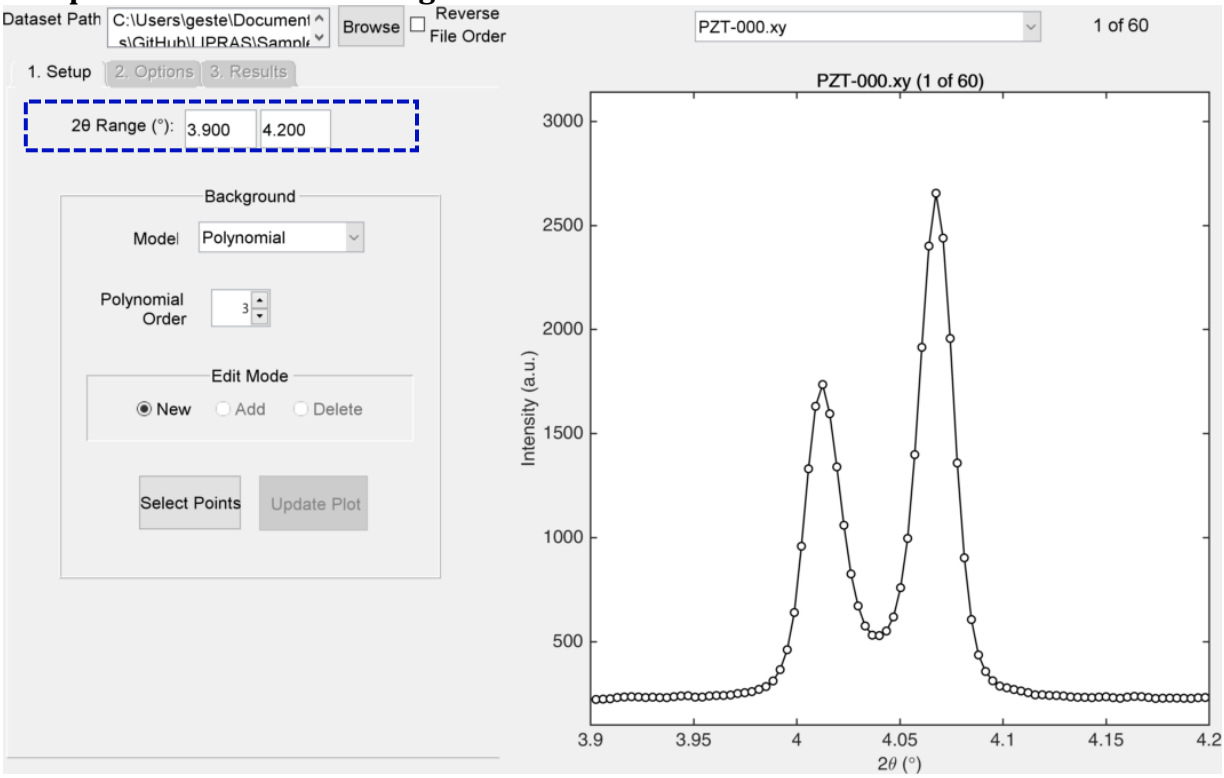
2. Setting the Fit Range and Selecting a Background

After loading a dataset, the plot window will show the first file that was loaded. All files can be viewed by selecting them individually in the drop-down menu, which is boxed in blue on the figure below.

Once Data is loaded, files can be viewed with the drop down mean (boxed in blue)



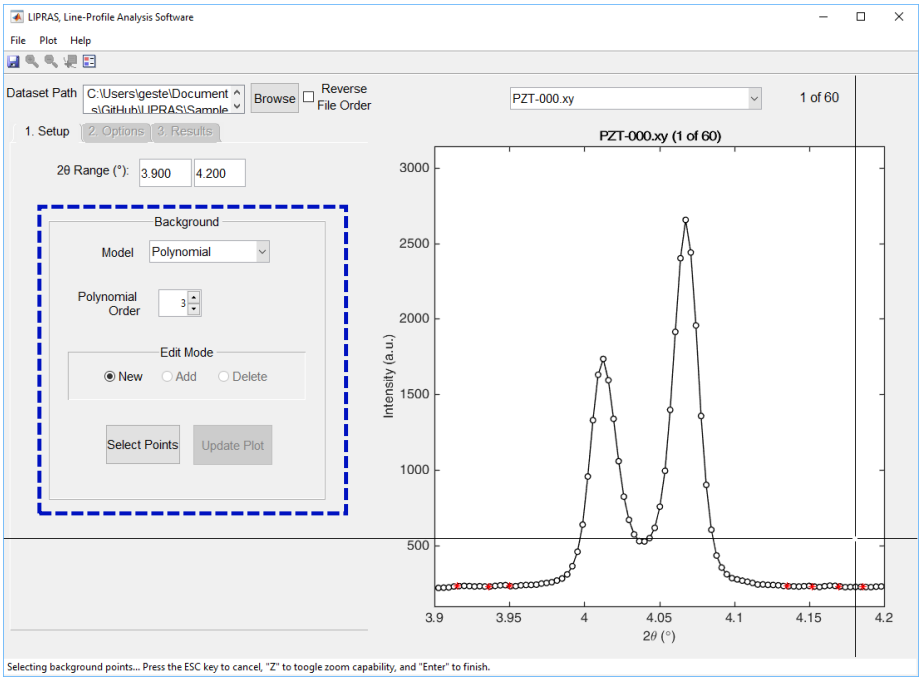
The '2θ Range' can be changed manually, this determines the bounds for the fit, in this example the bounds are changed to 3.9 and 4.2



Background can be modeled using a polynomial function or a spline function, with background points being user selected. To begin selecting a background, select the 'Select Points' button. The cursor will then turn into a crosshair and the user may begin selecting points that will be used to model the background. **Note:** The user needs to have more points than the number of polynomial orders. For example, for a polynomial order of 3 at least 4 background points need to be selected. Otherwise, the program will not continue and will prompt the user to select more points. To finish selecting points, press the 'Enter' key, and to Cancel and discard all points selected, press the 'Esc' key. Points can be added and deleted by toggling the 'Add' and 'Delete' buttons then clicking 'Select Points', but at least one point must be added or deleted before hitting 'Enter' or 'Esc'. Alternatively, the user can right-click at any time to delete a point as well. To zoom in while selecting a point, press the 'Z' key, after zooming into the desired range press the 'Esc' key to return to point selection. To view what the background would look like with other polynomial orders, change the polynomial order, and click on 'Update Plot'. This option helps when dealing with complex backgrounds or when not refining the background in the least squares routine. The figure below shows a

screenshot of the background point selection mode active, at the bottom left of the window there is a tooltip reminding the user to hit 'Enter' to continue or 'Esc' to discard the points.

Background points are selected by hitting 'Select Points'



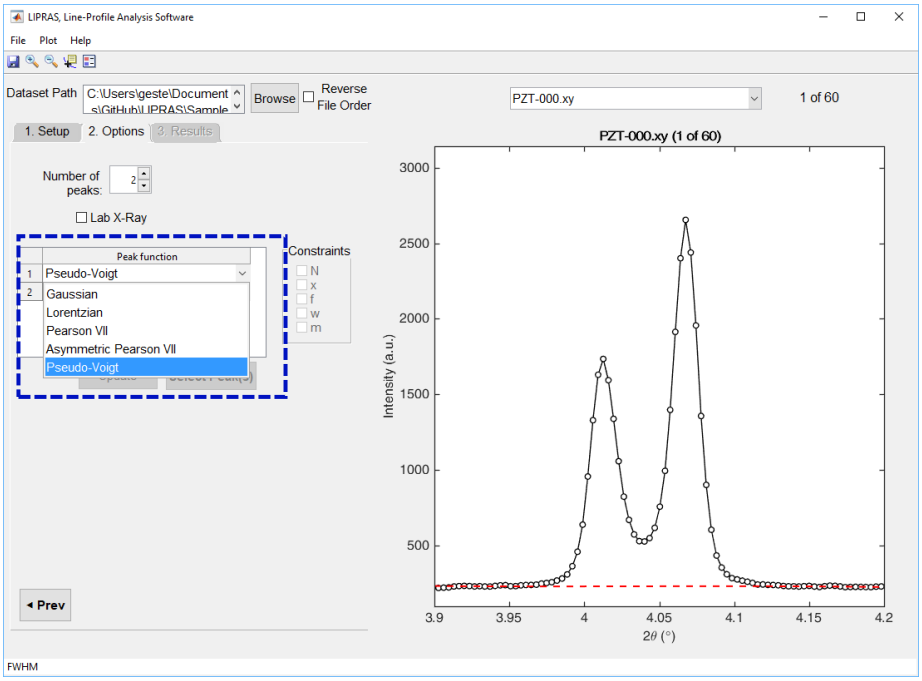
LIPRAS can also include the polynomial background in the least squares routine to refine the background and profile coefficients together. When using the polynomial option, LIPRAS takes advantage of centering and scaling using the average and standard deviation of the 2θ (x) values, see MATLAB's polyfit function for details. Thus, when the background model is refined in the least squares routine the x value is substituted with the Eq. 1**Error! Reference source not found.** below, where μ_x is the average of all the x values and σ_x the standard deviation.

$$x = \frac{x - \mu_x}{\sigma_x} \qquad 1$$

3. Peaks and Profile Functions

After selecting a background, the number of peaks need to be specified. For each peak introduced in the 'Number of peaks' tab, a profile function needs to be selected. To specify a profile function, click on the empty box that is created upon increasing the number of peaks. This is shown in the figure below.

Selecting Profile functions can be done via a drop-down menu, 5 profile functions are currently available



LIPRAS uses four well-known symmetric line-profile functions and one asymmetric function: Gaussian, Lorentzian, Pearson-VII, pseudo-Voigt, and Split Pearson-VII (asymmetric). The maximum number of peaks that can be applied is 20. Equations 2-5 describe the four symmetric functions in LIPRAS, and are also described in Ref. [1].

Gauss:

$$y(x) = G(x) = \frac{C_G^{1/2}}{\sqrt{\pi}H}\exp(-C_G x^2) \qquad 2$$

Lorentz:

$$y(x) = L(x) = \frac{C_L^{1/2}}{\pi H}(1 + C_L x^2)^{-1} \qquad 3$$

Pseudo-Voigt:

$$y(x) = PV(x) = \eta G(x) + (1 - \eta)L(x) \quad (4)$$

Pearson-VII:

$$y(x) = PVII(x) = \frac{\Gamma(\beta)}{\Gamma(\beta - 1/2)} \frac{C_P^{1/2}}{\sqrt{\pi}H} (1 + C_P x^2)^{-\beta} \quad (5)$$

- H is the full widths at half maximum (FWHM)
- $x = 2\theta i - 2\theta k/H_k$, Bragg angle of the *i*th point in the diffraction pattern with its origin at the position of *k*th peak divided by the peak's FWHM
- $2\theta_i$ and $2\theta_k$ are the Bragg angle of the *i*th point and the calculated (ideal) Bragg angle of the *k*th Bragg reflection
- C_G=4ln2, C_L=2, and C_p=4($2^{1/\beta}$-1)
- η is the pseudo-Voigt function mixing parameter, where η=1 represents a Gauss function and η=0 a Lorentz function
- Γ is the gamma function
- β exponent that describes the tails of the function, where β=1 is a Lorentz function and with increasing β the function resembles a Gauss function

Each function has its strength and weaknesses in terms of its intrinsic profile. The Gauss function is a symmetrical function and exhibits a Gaussian distribution of the diffraction intensity. The Lorentz function has more pronounced tails compared to the Gauss function. The pseudo-Voigt is a linear combination of the Gauss and Lorentz function and is the most widely used line-profile function used to fit diffraction data. The Pearson-VII function is like a Lorentz distribution, but its profile is altered by the parameter β. Depending on the value of β the profile can be more Lorentzian when closer to 1 and Gaussian when approaching a value of 10.

Peak asymmetry can be encountered in diffraction that is not related to instrumental contribution and is a signal from the sample. In this case, an asymmetric function is needed to fit the diffraction profile. One option, is a modification of the Pearson-VII function to allow both sides of the function to have different integrated intensities. This is called the asymmetric Pearson-VII which is taken from Ref. [2] and is shown in Eq. 6:

Asymmetric PVII:

$$y(x) = \begin{cases} \frac{N_L}{H_L\sqrt{\pi}} \frac{\Gamma(\beta_L)}{\Gamma(\beta_L - 1/2)} [1 + C_P x^2]^{-\beta_R} & for\ x < x_0 \\ \frac{N_R}{H_R\sqrt{\pi}} \frac{\Gamma(\beta_R)}{\Gamma(\beta_R - 1/2)} [1 + C_P x^2]^{-\beta_R} & for\ x \geq x_o \end{cases} \quad 6$$

where N_L and N_R are the integrated intensities of the left and right-hand side of the line-profile. The following equality must hold true in the case of x=x_0

$$\frac{N_L}{H_L\sqrt{\pi}} \frac{\Gamma(\beta_L)}{\Gamma(\beta_L - 1/2)} = \frac{N_R}{H_R\sqrt{\pi}} \frac{\Gamma(\beta_R)}{\Gamma(\beta_R - 1/2)} \quad 7$$

Rearranging gives one variable as a function of the others:

$$H_R = H_L \frac{N_R}{H_L} \frac{\Gamma(\beta_R)}{\Gamma(\beta_R - 1/2)} \frac{(\beta_L - 1/2)}{\Gamma(\beta_L)} \quad 8$$

The total integrated intensity of the distribution (I_{hkl}) is given as

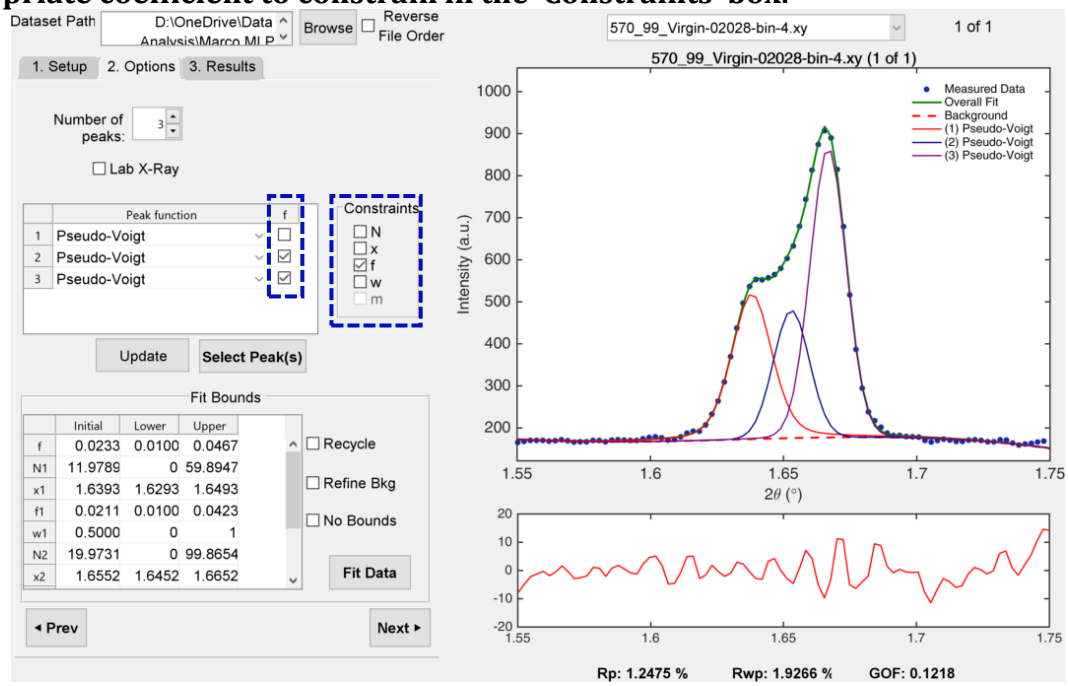
$$I_{hkl} = \frac{N_L + N_R}{2} \quad 9$$

If the data was measured with a laboratory X-ray diffractometer, the respective Cu-Kα2 contributions from each Bragg reflection can be calculated and accounted for in the fit. Files with the extension XRDML automatically activate this feature of LIPRAS and read in the corresponding Cu-Kα1 and Kα2 wavelengths from the file. However, if the files are not XRDML or there is not Kα2 component this feature can be manually activated/deactivated.

Constraining Profile Coefficients

One of the strong suits of LIPRAS is its ability to constrain profile coefficients to be equal. For example, an analysis can be conducted in which the FWHM of two profiles are constrained to be equal. This becomes useful when fitting overlapped Bragg reflections or when mixing coefficients of two pseudo-Voigt or Pearson-VII functions reach an upper or lower bound in the least squares routine. To activate the constraints, select the coefficient to constraint in the constraints panel highlighted in the figure below. To constraint profile coefficients at least two functions must be used, and when using more than 2 profile functions a separate column of checkboxes will appear to select which profile functions the user would like to constrain as shown in the figure below.

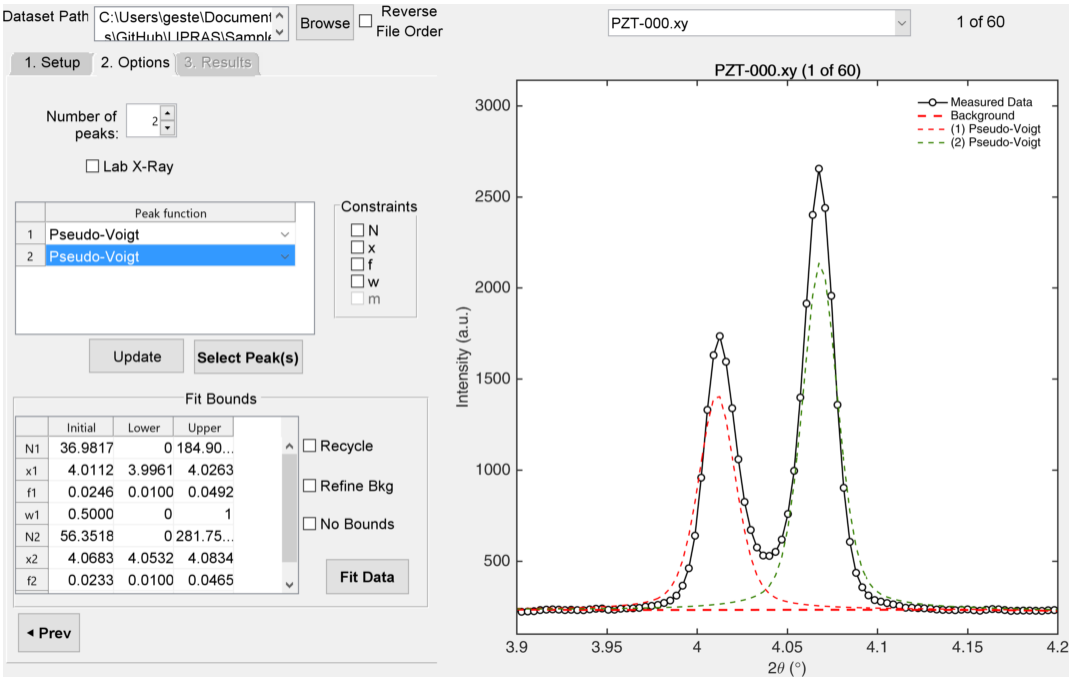
Constraining profile coefficients to be equal can be initiated by checking the appropriate coefficient to constrain in the 'Constraints' box.



4. Fit Bounds and Options

After peak position(s) have been selected, initial guesses are generated from the peak position(s). The selected profile functions are plotted with the initial guesses in dashed lines. Before starting the fit, starting values for the selected peaks can be adjusted to ensure that the starting parameters are relatively close to the anticipated final solution. After the table in 'Fit Bounds' has been populated, a fit can begin by hitting 'Fit Data'. There are three checkboxes contained within this area of the GUI that allow for more customizable fitting options: Recycle, Refine Bkg, and No Bounds.

After selecting peak positions, estimates are generated and shown in dashed lines for the profile functions used, which can be adjusted



Recycle

Recycle is a very powerful feature in LIPRAS. When toggled, this feature uses the initial parameters in the table within 'Fit Bounds' and applies them to the first data file. Then, after a least squares solution is reached for the first file, these results become the starting parameters for the second file. The method repeats until all files have been fitted. This option is recommended when fitting diffraction patterns that have incremental changes with

applied stimulus. If this option is not check the values within the 'Fit Bounds' table will be used for every diffraction pattern.

Refine Bkg

Refine Bkg allows the polynomial background model to be included in the least squares routine along with the profile functions selected. The polynomial order is set by the user in the 'Setup' tab. If this option is not checked than the background is determined from the user selected points and subtracted from the raw data before starting the least squares routine. Refining the background with the profile functions typically provides better goodness-of-fit. Therefore, this method is recommended. Typically, this option can be checked with the 'Recycle', but for best results it is suggested to run an initial fit, then hit 'Update' to allow the Starting Parameters to be updated with the results of the previous analysis, then checking 'Refine Bkg' and conducting another fit. More details in the subsequent sections.

No Background in Least Squares: 'Refine Bkg' is <u>NOT</u> Checked,

By considering the intensity, $Y(i)$, of the ith point (where i ranges from 1 to n, n is the total number of measured points) of the diffraction profile, Eq. 10 can be used to model an observed diffraction profile.

$$Y(i) = \sum_{k=1}^{m} I_k[y_k(x_k)] \qquad 10$$

In Eq. 10, I_k is the intensity of kth Bragg reflection, and is dependent on the profile function used to model the pattern y_k whose input is $x_k=2\theta_i-2\theta_k$. The use of a Bragg intensity(I_k) as a multiplier in Eq. 10 allows for analyzing diffraction profiles using normalized functions that are independent of peak intensity. This means that the definite integral of the line-profile function $y_k(x_k)$ integrates to unity, *i.e.* $\int_{-\infty}^{\infty} y_k(x_k)dx = 1$. [1] In this approach, the background is calculated based on the points picked by the user. Once the background is fitted, based on the points, and evaluated, the inputted data is subtracted from the background and then the model is applied. In this method, the refinement of the background does not take place and this can be used to see clearer trends in the coefficients of the profile function without influence of the background since it's not refined simultaneously with the profile model.

Background in Least Squares: 'Refine Bkg' IS Checked,

In this method, the background model $b(i)$ is simultaneously refined with the profile model, as shown in Eq. 11. This approach results in better fits than when the background model is excluded from least squares. However, the profile coefficients are dependent on the fit of the background, and in weak signal data or data with complex background, the profile coefficients can fall into false minimum due to the background accounting for the intensity of a peak.

$$Y(i) = b(i) + \sum_{k=1}^{m} I_k[y_k(x_k)]\,; \qquad 11$$

No Bounds

The 'No Bounds' options, when checked, ignore the upper and lower limits specified and only takes the starting parameters as an input. This activates a true Levenberg–Marquardt least squares algorithm to take place instead of the Trust-Region algorithm, which is the default when using upper and lower bound limits. This option should be used with caution when using profile shape functions with mixing coefficients such as pseudo-Voigt and Pearson VII since it can lead to unphysical values or the solution may diverge causing an error. Typically, it is best if the user starts out by refining the profile coefficients first, hitting 'Update', then checking 'Refine Bkg', fitting the data again, hitting 'Update', then select 'No Bounds' while 'Refine Bkg' is checked and hit 'Fit Data'.

5. Fitting Data

To minimize the applied model to the actual data, LIPRAS uses a least squares routine that is part of the Curve Fitting Toolbox™ developed by MathWorks®. This least squares fitting method minimizes the difference between the measured diffraction peak and the modeled diffraction profile using least squares minimization. In this method, errors in the parameters are usually reported as confidence intervals for the estimated parameter values. There are two minimization routines that are used by the Curve Fitting Toolbox™ which depend on whether the coefficients have specified upper and lower bounds or they do not. When the reduction algorithm must abide by the users upper and lower bounds, a Trust-

Region-Reflective Least Squares algorithm is used, and when 'No Bounds' is selected the Levenberg-Marquardt Method is used. For more information on these reduction algorithms **click here**.

The reduction algorithm used by LIPRAS allows for the input of weights which means the minimization routine can take errors of the measured intensity into account in the least squares refinement. When errors are properly read in, or when the 'Weights' are defined in the Preference tab as anything but 'None' the least squares algorithm minimize χ^2 since $w=1/\sigma_i^2$ and σ_i is usually the square root of the observed intensity of the ith point.

$$\chi^2 = \sum \frac{(y_{obs} - y_{calc})^2}{{\sigma_i}^2} = \sum w(y_{obs} - y_{calc})^2 \qquad 12$$

To evaluate goodness-of-fit, by default the Curve Fitting Toolbox™ use the following: R-square, adjusted R-square, and root mean squared error. Additionally, LIPRAS also uses statistical metrics that are commonly found in some Rietveld programs such as: R_p, R_{wp}, and reduced chi-square, which are described in the below.

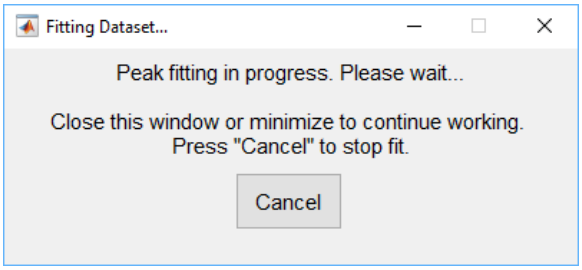
$$R_p = \left(\frac{\sum|y_{obs} - y_{calc}|}{\sum y_{obs}}\right) \times 100 \qquad 13$$

$$R_{wp} = \left[\frac{\sum w(y_{obs} - y_{calc})^2}{\sum w(y_{obs})^2}\right]^{1/2} \times 100 \qquad 14$$

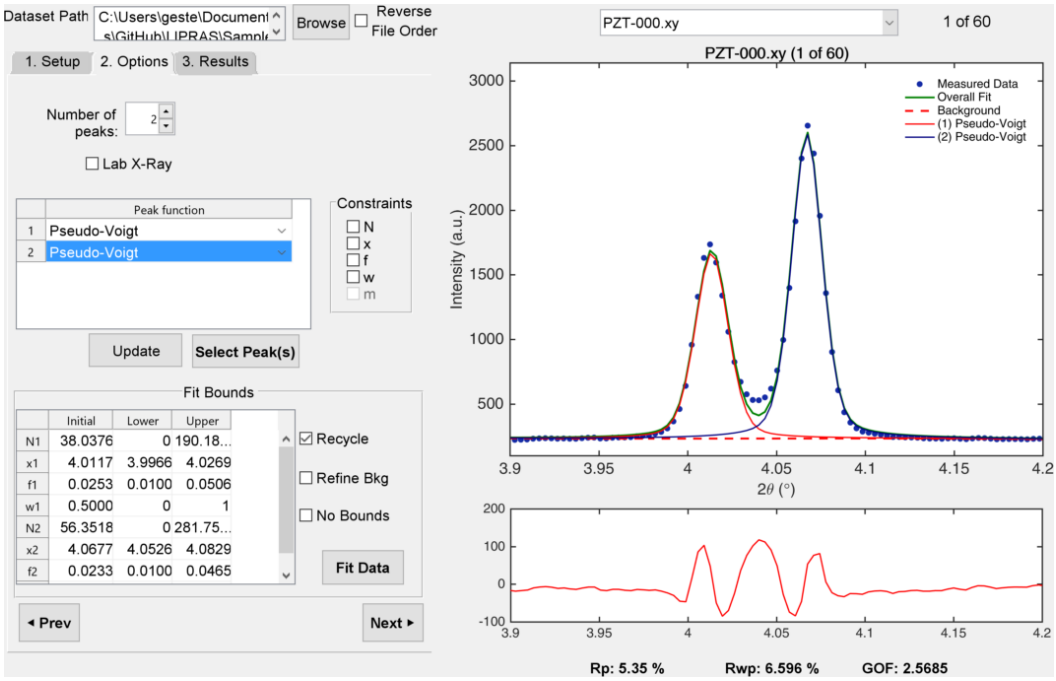
$$\chi^2_{red} = \frac{\chi^2}{v} = \text{GOF}$$

$$v = \#\ of\ data\ points - \#\ of\ variables \qquad 15$$

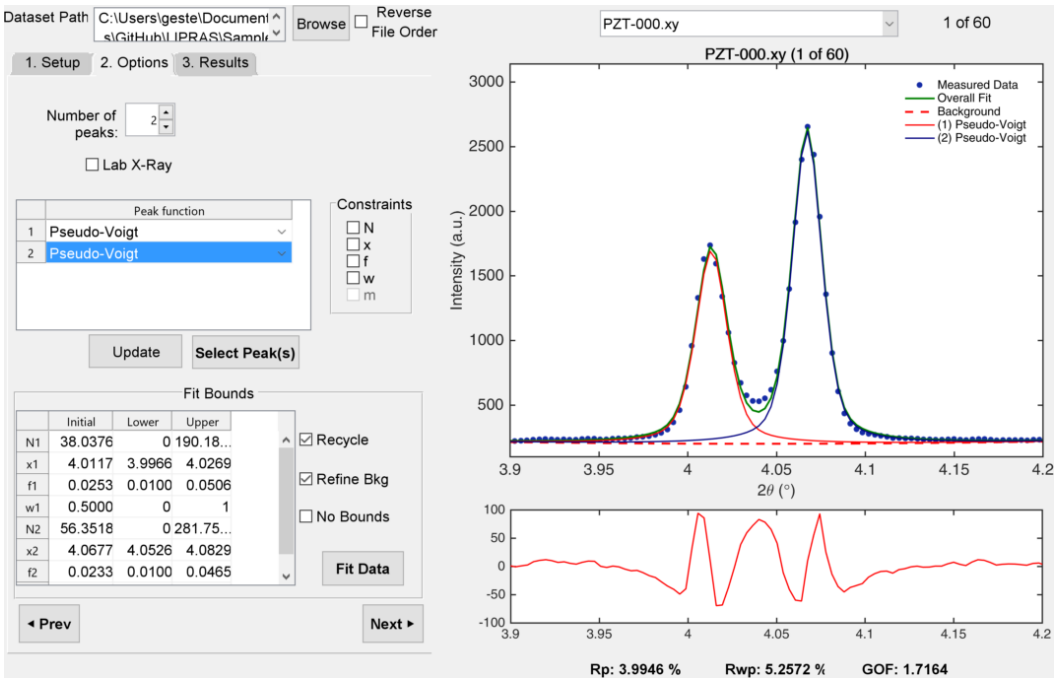
After hitting 'Fit Data' a dialog box will appear stating the fit is in progress



The resulting fit will automatically display in the GUI window after a fit has finished



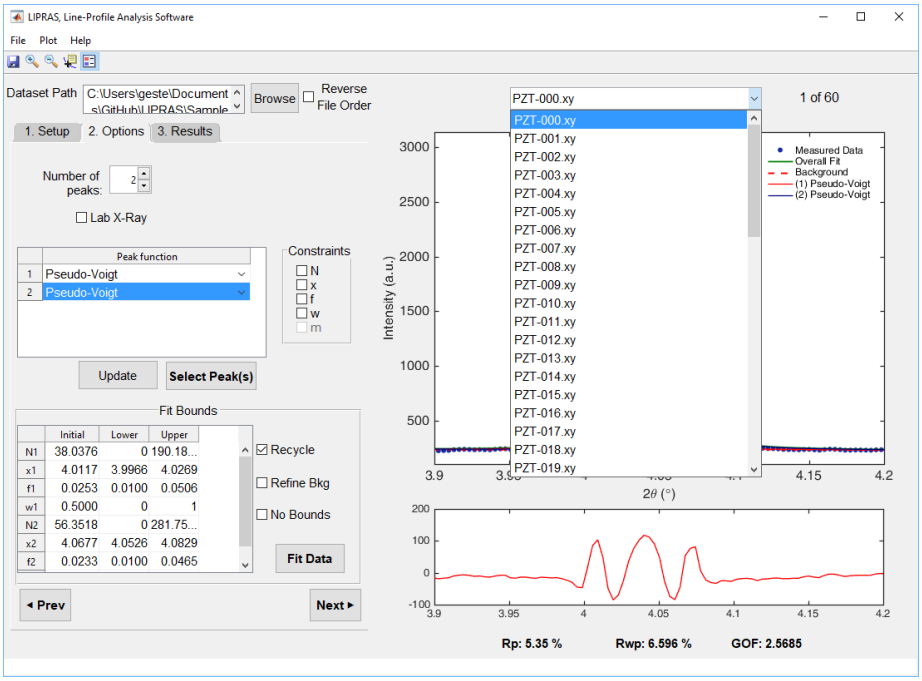
Fit conducted with the background included in the least squares routine



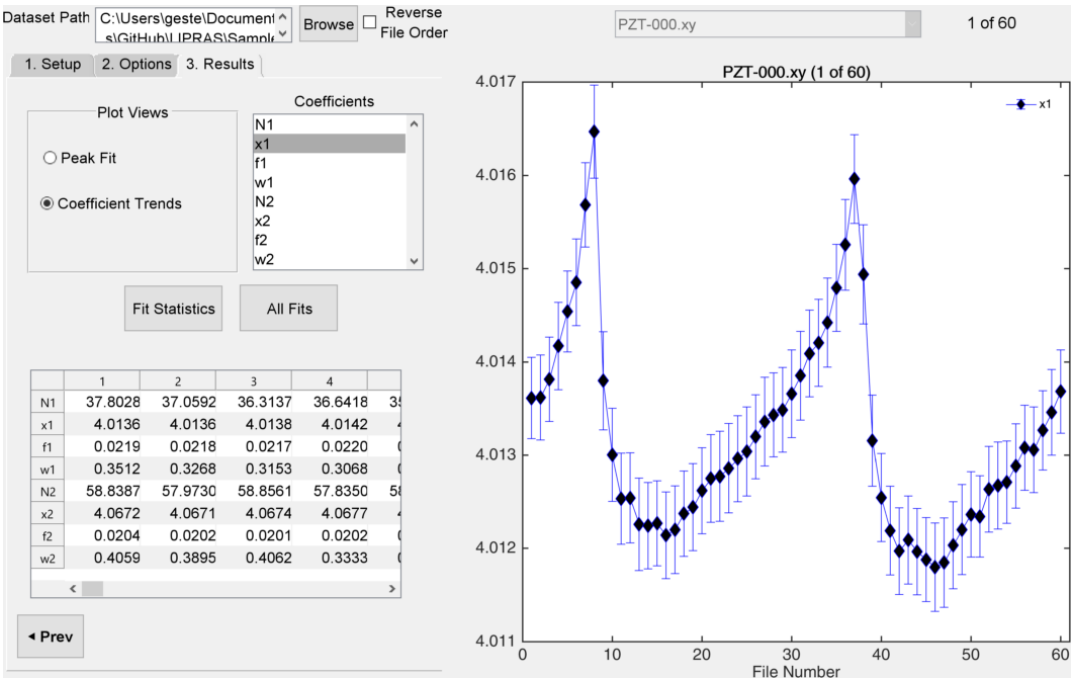
6. Viewing Results

After a fit, the 'Results' tab will become available, which allows the user to view the resulting fit and coefficients as a function of file number. This allows the user to quickly judge the fit and adjust their model or starting parameters. The 'All Fits' button will plot all the fits in one window, but this option should be used with low file numbers, *i.e.* less than 20 files. Lastly, the 'Fit Statistics' button plots the goodness-of-fit statistics as a function of file number, this includes: R-squared, adjusted R-squared, root mean squared error, R_p, R_{wp}, and χ^2_{red}.

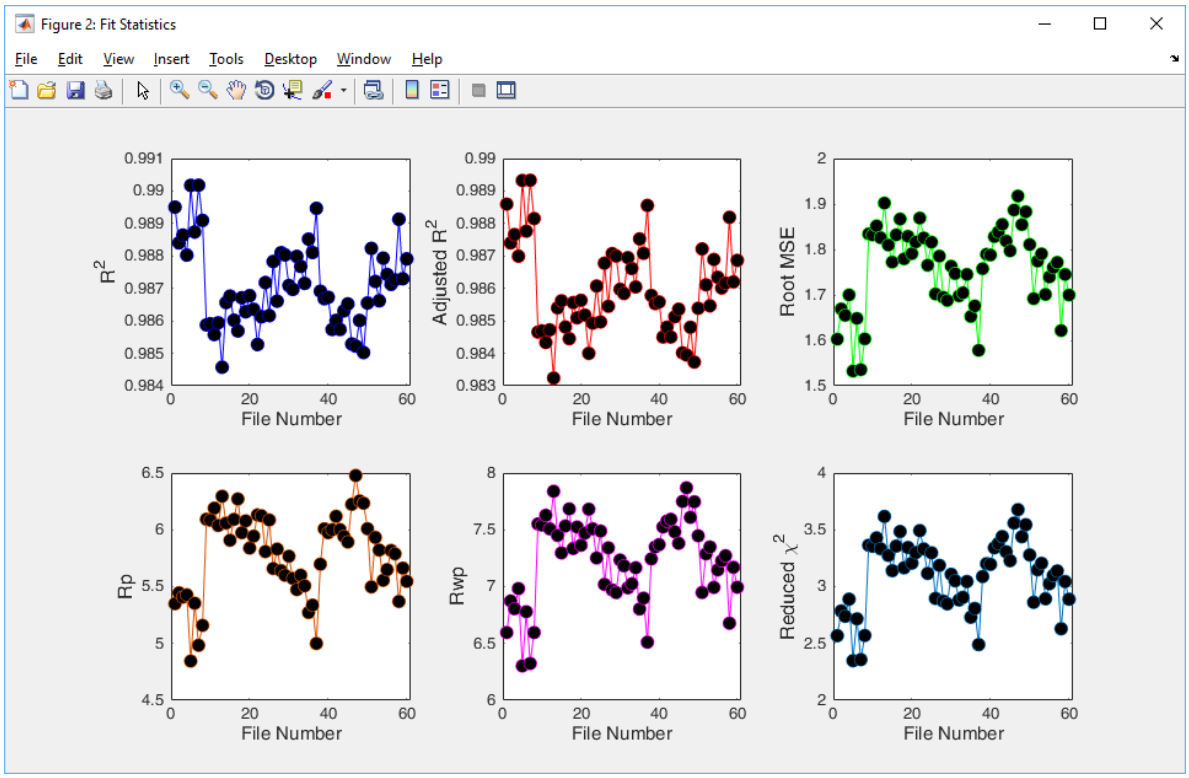
To view the resulting fit for other files, select them from the drop-down menu or click on the 'Results' tab



Viewing resulting profile coefficients as function of file number can be achieved by clicking on 'Coefficient Trends' in the 'Results' tab after conducting a fit



Fit Statistics



7. Exporting Results: Fmodel and Fdata Files

Once a fit has been completed, the resulting profile coefficients and data are saved in the directory where the raw data is located. To view the resulting profile coefficients outside of LIPRAS, there is an Fmodel file containing the resulting profile coefficients for all files that were fit in the analysis. An example of an Fmodel file is shown in the figure below. Each Fmodel file will contain the resulting profile coefficients and error. The error is calculated using the 95% confidence interval and subtracting it from the resulting value. Other useful goodness-of-fit statistics are also printed in this file such as R-squared, Adjusted R-squared, root mean squared error, R_p, R_{wp}, and reduced χ^2 (χ^2_{red}). To open this file, it is recommended to use a text editor. The authors recommend **Notepad++** for Windows users.

Example of an Fmodel File

```
This is an output file from a MATLAB routine.
The following peaks are all of the type: Pseudo-Voigt; Pseudo-Voigt;

FileName    N1  N1_Error    x1  x1_Error    f1  f1_Error    w1  w1_Error    N2  N2_Error
PZT-000_xy  23.34989    1.24188 1.62403 0.00022 0.01619 0.00070 0.53461 0.14240 23.51645
PZT-001_xy  23.19547    1.23743 1.62385 0.00022 0.01594 0.00072 0.56641 0.14226 24.30896
PZT-002_xy  21.68311    1.23785 1.62402 0.00023 0.01630 0.00070 0.44673 0.15627 23.49282
PZT-003_xy  21.01769    1.05110 1.62380 0.00019 0.01617 0.00060 0.41716 0.13806 24.40783
PZT-004_xy  21.99913    1.24187 1.62393 0.00023 0.01640 0.00074 0.52092 0.15114 24.37067
PZT-005_xy  20.73420    1.08835 1.62398 0.00021 0.01642 0.00067 0.47747 0.14238 24.07851
PZT-006_xy  20.03252    1.15473 1.62402 0.00024 0.01645 0.00079 0.56391 0.15276 24.79723
PZT-007_xy  20.17317    1.41575 1.62416 0.00031 0.01760 0.00100 0.55137 0.18524 26.92044
PZT-008_xy  25.94919    1.58788 1.62485 0.00025 0.01712 0.00077 0.46877 0.16364 22.59879
PZT-009_xy  28.01257    1.54085 1.62485 0.00022 0.01635 0.00068 0.49842 0.14474 21.25452
PZT-010_xy  27.79291    1.51445 1.62464 0.00021 0.01584 0.00066 0.49180 0.14510 20.95588
PZT-011_xy  29.11020    1.50519 1.62455 0.00020 0.01560 0.00064 0.52923 0.13717 21.47010

```

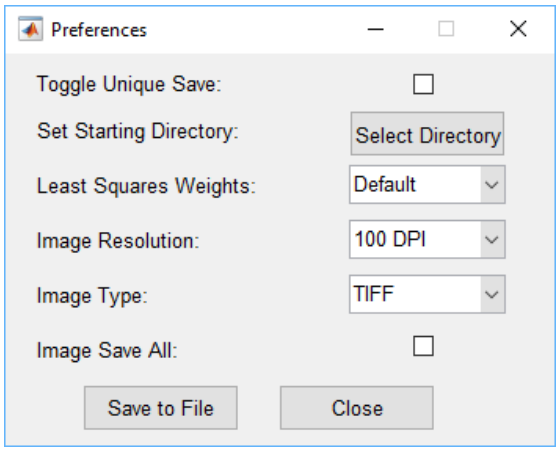
The Fdata file contains the raw data and the evaluated profile function with individual peaks. An example of an Fdata file is shown in the figure below. The columns that are printed with all Fdata files are 2θ, Obs, Calc, BkgdFit, Weights, and Peaks. The peaks column are the evaluated profile functions with the resulting coefficients. The sole purpose of the Fdata file is to create high quality figures of the resulting fits for posters, presentations, and publications.

Example of a Fdata File

```
This is an output file from a MATLAB routine.
All single peak data (column 3+) does not include background intensity.

2theta    Obs       Calc     BkgdFit      Weights      Peak1    Peak2
3.55055550  1810.39820000   1824.30867468   1807.27886734   0.00055236  0.00000000  17.02980734
3.55359890  1829.17380000   1829.30231278   1811.78516262   0.00054669  0.00000000  17.51715016
3.55664230  1815.27430000   1834.29040221   1816.26472726   0.00055088  0.00000000  18.02567495
3.55968570  1800.21950000   1839.27424646   1820.71762166   0.00055549  0.00000000  18.55662479
3.56272910  1816.33750000   1844.25524136   1825.14390623   0.00055056  0.00000001  19.11133512
3.56577230  1827.69670000   1849.23455624   1829.54335310   0.00054714  0.00000003  19.69120311
3.56881570  1837.92810000   1854.21445196   1833.91660095   0.00054409  0.00000007  20.29785094
3.57185910  1860.94700000   1859.19632790   1838.26342016   0.00053736  0.00000017  20.93290756
3.57490250  1880.33980000   1864.18204226   1842.58387115   0.00053182  0.00000043  21.59817068
3.57794590  1910.09060000   1869.17359751   1846.87801430   0.00052354  0.00000106  22.29558215
3.58098940  1905.28640000   1874.17331902   1851.14604984   0.00052486  0.00000255  23.02726663
3.58403280  1872.72140000   1879.18321371   1855.38775769   0.00053398  0.00000600  23.79545002
3.58707620  1850.98190000   1884.20597387   1859.60333891   0.00054025  0.00001389  24.60262107
3.59011940  1873.00680000   1889.24400987   1863.79257946   0.00053390  0.00003155  25.45139885
3.59316280  1884.53600000   1894.30096471   1867.95609036   0.00053063  0.00007032  26.34480403
3.59620620  1883.18990000   1899.37975878   1872.09365585   0.00053101  0.00015380  27.28594913
```

8. Preferences Menu



Toggle Unique Save: Creates a new folder every time 'Fit Data' is hit

Set Starting Directory: Set the starting directory when browsing for data

Least Squares Weights: Weights involved in least squares routine, as shown **here**. The 'Default' option uses the errors from files that have them. Otherwise the Weight defaults to '1/obs' if the file does not have errors included.

Image Resolution: Sets the DPI for images that are exported.

Image Type: Sets the image type for exported images

Image Save All: Saves all fits that were conducted, by default LIPRAS will only save the first figure unless this is checked.

9. Input/Output Parameter File

LIPRAS has the option to write an output parameter file that contains the necessary information to recreate the resulting fit. This file can be generated by going to File>Export> Save Fit Parameter file or by hitting CTR+S. This very same file can be imported into LIPRAS after loading data in to recreate and re-populate what was done previously. All output parameter files are of .txt extension and contain the following information as shown in the figure below:

1. 2θ range
2. Background Model
3. Polynomial Order
4. Background Points
5. Cu-Kalpha 1,2
6. Fit Fxns
7. Constraints
8. Peak Positions
9. Initial Fit Parameters
10. Refine Background Coefficients (only if refined)

Example of an Output Parameter File

```
2ThetaRange: 11.50000 11.85000

BackgroundModel: Polynomial
PolynomialOrder: 2
BackgroundPoints: 11.502260 11.510655 11.513884 11.518404 11.796079 11.810932 11.820618 11.832242 11.840637 11.846448
Cu-KAlpha1: n/a

Cu-KAlpha2: n/a

FitFunction(s):
Pearson VII; Pearson VII; Pearson VII;
Constraints: {'m'} {'m'} {'m'}
PeakPosition(s): 11.621303 11.647181 11.667884

== Initial Fit Parameters ==
m N1 x1 f1 N2 x2 f2 N3 x3 f3
SP: 1.50000 93.34307 11.62130 0.03045 80.00000 11.64718 0.04169 154.83736 11.66788 0.04047
LB: 0.50000 0.00000 11.60380 0.01000 0.00000 11.62968 0.01000 0.00000 11.65038 0.01000
UB: 20.00000 466.71537 11.63880 0.06090 797.53797 11.66468 0.08339 774.18679 11.68538 0.08094

== Bkg Coeffs ==
SP: -42.48338 -35.96757 183.37026
```

Bayesian Inference with LIPRAS

A Bayesian inference analysis can be carried out on the resulting least squares results to generate a full description of the errors for all profile parameters. **MATLAB users need the Statistics and Machine Learning Toolbox** to conduct this analysis. This is an alpha release and may contain errors. Works best on symmetrical functions, asymmetric function can be a hit or miss. The algorithm used is a Markov Chain Monte Carlo (MCMC) Metropolis-in-Gibbs, which is described in the following paper: **Bayesian** and is now included in every LIPRAS download.

Description of Model

Scattered intensity (I) is typically measured as a function of scattering angle (2θ), resulting in Bragg peaks

$$I = f(2\theta|\alpha) \qquad 1$$

f is a profile shape function and α is a set of parameters for the profile shape function, which in this case is a pseudo-Voigt, but can be any of the functions listed in **Peaks and Profile Functions**.

$$f(2\theta|\alpha) = \frac{wI\sqrt{4ln2}}{\sqrt{\pi}F} exp\left(-\frac{(4ln2)(2\theta-2\theta_0)^2}{F^2}\right) + 2\frac{I(1-w)}{\pi F}\left(1+\frac{4(2\theta-2\theta_0)^2}{F^2}\right)^{-1} \qquad 2$$

New parameters are generated by $N(\alpha_{st}, s^2)$ which is the normal distribution with mean α_{st} and standard deviation s.

$$\alpha_{new} = N(\alpha_{st}, s^2) \qquad 3$$

Bayes Theorem

$$P(\alpha|data) = \frac{P(data|\alpha)P(\alpha)}{P(data)} \quad (4)$$

α is the parameter of interest

$P(\alpha|\text{data})$ is the posterior distribution (probability of parameter values given the data)

$P(\text{data}|\alpha)$ is the likelihood (probability of data given the parameter values)

$P(\alpha)$ is the prior distribution

$P(\text{data})$ is the marginal likelihood

$$P(data|\alpha) = (2\pi\sigma^2)^{-n/2}\exp\left(-\frac{1}{2\sigma^2}\sum_{j=1}^{n}(I_{obs,j} - I_{cal,j})^2\right) \quad (5)$$

$I_{obs,j}$ and $I_{cal,j}$ is the calculated intensity, n is the number of data points and σ^2 is the variance

Algorithm the MATLAB Should Follow:

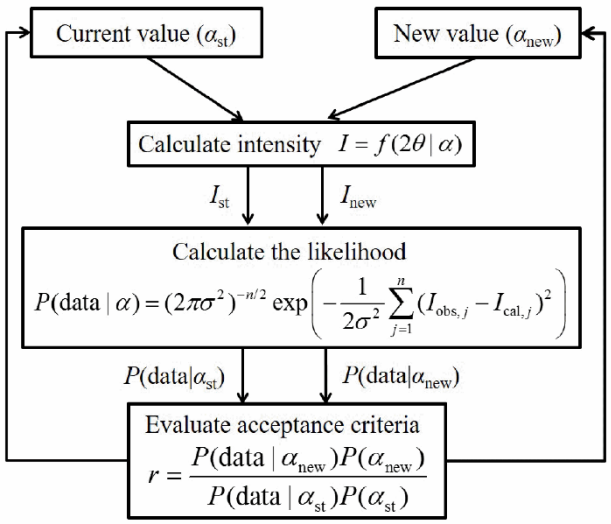
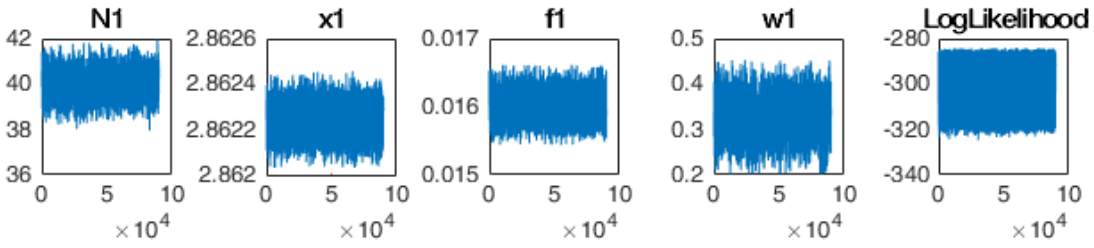


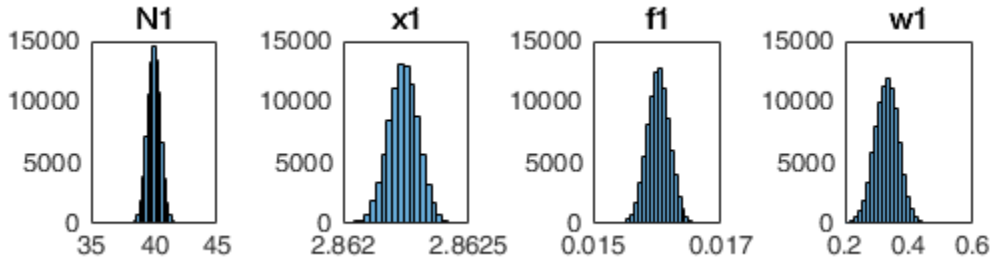
Figure is from the following publication: **Bayesian**

Using Bayesian GUI

- Launch Bayesian GUI
- Table should be populated from results of least squares refinement, you can change these defaults by switching to 'Custom Bounds'. To reset any changes, double-click the 'Default Bounds'.
    - Note: If you did peak fitting of multiple files you will do Bayesian inference on each file. The default parameters are taken individually from each least squares result of each file. If you use 'Custom Bounds' you will apply these bounds to **ALL** files. 'Custom Bounds' is best used for single file purpose.
    - *Upper bound and lower bound* are determine by taking the difference between the mean value (starting parameter) and the Error*4. The error is just the resulting value minus the reported confidence interval. The multiplier can be changed using the 'Bounds Multiplier'.
    - *Standard Deviation for parameters* are calculated from the 95% confidence interval generated from least squares results and a divider can be selected to enlarge or shrink these for all parameters in the table by using the 'SD Divider'.
- If you have no idea what you are doing, click 'Auto' and hit 'Run Bayesian'
- In the end, you want all your trace plots for the coefficients to resemble something like the following:



- You can tune σ^2 to determine how log(likelihood) behaves. LogLikelihood is just the log(Eq. 5). However, the important thing is to have all your coefficients oscillating around one value. When looking at the histogram for each coefficient plotted above, you get the following, which describes your posterior distribution of all your parameters:



- From here it is easier to see what your mean value is and its corresponding sigma.
- All of these results are saved in .BAYES files which contain the fit_mean, the final over all fit, along with its high and low intervals, trace plots of all coefficients, and log(likelihood) which is used to recreate plots such as those shown above.
- The 'Plot Bayesian Fit' allows you to view the resulting Bayesian fit and compare it to the least squares fit with calculated Rp values for each fit.

Acknowledging and Citing LIPRAS

If you use LIPRAS in your research, please cite it. There is a DOI that has been generated for LIPRAS on ResearchGate, DOI: 10.13140/RG.2.2.29970.25282/3, **click here** to access the technical document associated with this DOI. Citing this document is one of the ways we can measure its success. Lastly, you are welcome to use any of the two following logos for presentations or posters.

Citation, use of the following:

1. Esteves, G., Ramos, K., Fancher, C. M., and Jones, J. L. LIPRAS: Line-Profile Analysis Software. (2017). DOI: 10.13140/RG.2.2.29970.25282/3.

2. Esteves, G., Ramos, K., Fancher, C. M., and Jones, J. L. LIPRAS: Line-Profile Analysis Software. (2017). https://github.com/SneakySnail/LIPRAS

LIPRAS Logo

Obtaining LIPRAS and Updates

LIPRAS is distributed through three sources: MATLAB File Exchange and GitHub for those who wish to use LIPRAS within the MATLAB environment, and SourceForge for those who want a standalone version with no MATLAB dependency. Links are provided by clicking on the bolded text below this paragraph. LIPRAS is open source under a BSD License, which is stated at the beginning of this document. To view the full code, visit the GitHub repository for LIPRAS which is where the code is maintained and updated. To check for updates while using LIPRAS, go to Help> Check for Updates. This prompt LIPRAS to read the HTML page of the GitHub repository and make sure that the version number specified within LIPRAS matches the number of commits in the GitHub repository. If it does, then LIPRAS is up to date. Otherwise, it will prompt the user to go download the latest version by providing the link.

MATLAB Version: [Click Here]
Disclosure: Requires MATLAB® 2016b**,** or higher, with Curve Fitting Toolbox™ and **GUI Layout Toolbox.** Bayesian inference feature requires Statistics and Machine Learning Toolbox™.

Standalone Version: [Click Here]
Disclosure: It utilizes MATLAB® runtime which is free and can be downloaded online or through the installation manager.

References

[1] V. K. Pecharsky and P. Zavalij, *Fundamentals of Powder Diffraction and Structural Characterization of Materials*. Boston, MA: Springer US, 2009.

[2] J. E. Daniels, J. L. Jones, and T. R. Finlayson, "Characterization of domain structures from diffraction profiles in tetragonal ferroelastic ceramics," *J. Phys. D. Appl. Phys.*, vol. 39, no. 24, pp. 5294–5299, Dec. 2006.