Motofit

February 10, 2012

http://motofit.sourceforge.net
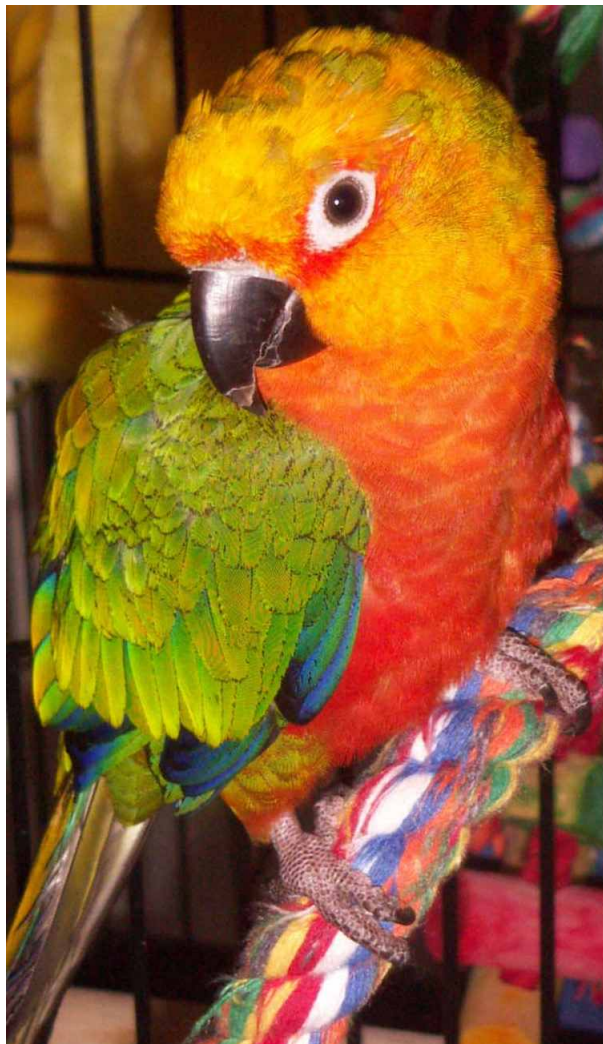Andrew_Nelson@users.sourceforge.net

Abstract

Motofit uses the Abeles Formalism to fit multiple contrast neutron and X-ray reflectometry data.

Contents

1 *Motofit*

Motofit aids the fitting of specular X-ray and neutron reflectivity data. It works within the analysis package IGOR PRO (Wavemetrics, OR). It was created in version 6 of IGOR.

The specular reflectivity is calculated using the *Abeles[1]* formulation (giving identical results to *Parratts[5]* recursion formula for stratified thin films), as a function of the perpendicular momentum transfer, Q_Z.

$$Q_z = \frac{4\pi}{\lambda} \sin\theta \tag{1}$$

Specular reflectivity (R) is defined as the ratio of reflected intensity over incident intensity, where the angle of reflection (θ) is equal to the angle of incidence.

Motofit was created with the aim of allowing a scientist to easily fit her/his reflectivity data. In particular, an easy way to analyse multiple contrast NR and XR data. This is achieved through a simple GUI, and genetic optimisation for fitting (Genetic Optimisation allows the use of initial guesses that are far from a final solution, which would otherwise trouble normal least squares fitting packages). Other reflectivity packages are obviously out there, but sometimes they don't allow you to do what you want.

A further advantage with *Motofit*/IGOR is that the graphics and output is publication quality, you just put the graph straight into your paper. You can control all aspects of the graphs you create. With most other analysis packages you have to export data into another plotting package, which take time. If you have to re-analyse data you have to do the export again. With *Motofit*, you just re-analyse, and the graphs are changed automatically. You can also save the whole experiment you are working on.

Motofit is also open source, which means you can change the behaviour of the program to whatever you want, if you don't like it. IGOR is a proven data-analysis program, which means you can rely on the numbers coming out

2 Acknowledging the use of *Motofit*

If you use *Motofit* to analyse data that you then use in a publication, then please cite *Motofit* in the references. This is so I can justify the time I spent writing the program to my bosses.

Inclusion in the reference should be done by referring to the *Journal of Applied Crystallography* paper[3].

A. Nelson. 'Co-refinement of multiple contrast neutron / X-ray reflectivity data using MOTOFIT.' *Journal of Applied Crystallography,* **39**, 273-276, 2006.

3 Fitting reflection from a stratified medium

The reflectivity is calculated using the *Abeles* matrix method for stratified interface[1].

The measured reflectivity depends on the variation in the scattering length density (SLD) profile, ($\rho(z)$) perpendicular to the interface. Although the scattering length density profile is normally a continuously varying function, the interfacial structure can often be well approximated by a slab model in which layers of thickness (d_n), scattering length density (ρ_n) and roughness ($\sigma_{n,n+1}$) are sandwiched between the super- and sub-phases. One then uses a refinement procedure to minimise the differences between the theoretical and measured reflectivity curves, by changing the parameters that describe each layer.

In this description the interface is split into n layers. Since the incident neutron beam is refracted by each of the layers the wavevector, k, in layer n, is given by:

$$k_n = \sqrt{k_0^2 - 4\pi(\rho_n - \rho_0)} \tag{2}$$

where $k_0 = Q/2$. Note that the wavevector can be complex if $4\pi(\rho_n - \rho_0)$ is greater than k_0^2. The Fresnel reflection coefficient between layer n and *n+1* is then given by:
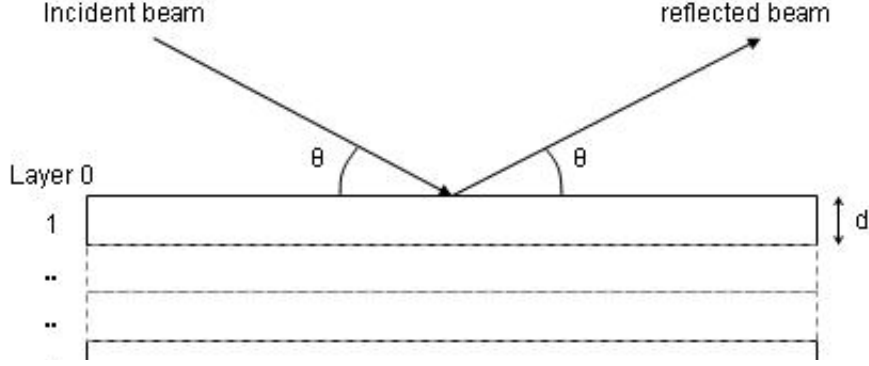
Figure 1: Reflection from a stratified medium

$$r_{n,n+1} = \frac{k_n - k_{n+1}}{k_n + k_{n+1}} \tag{3}$$

Since the interface between each layer is unlikely to be perfectly smooth the roughness/diffuseness of each interface modifies the Fresnel coefficient 4 and is accounted for by an error function, as described by Nevot and Croce[4].

$$r_{n,n+1} = \frac{k_n - k_{n+1}}{k_n + k_{n+1}} \exp\left(-2k_n k_{n+1} \sigma^2_{n,n+1}\right) \tag{4}$$

A phase factor, β is introduced, which accounts for the thickness of each layer.

$$\beta_n = k_n d_n \tag{5}$$

A characteristic matrix,cn is then calculated for each layer.

$$c_n = \left[\begin{array}{cc} \exp\left(\beta_n\right) & r_n \exp\left(\beta_n\right) \\ r_n \exp\left(-\beta_n\right) & \exp\left(-\beta_n\right) \end{array} \right] \tag{6}$$

The resultant matrix is defined as the product of these characteristic matrices, from which the reflectivity is calculated.

$$M = \prod_0^n c_n \tag{7}$$

$$R = \left| \frac{M_{00}}{M_{10}} \right|^2 \tag{8}$$

4 What files do you need?

1. Motofit_all_at_once.ipf

2. Motofit_Globalfit 2.ipf

3. Motofit_Batch.ipf

4. Motofit_SLDcalc.ipf

5. Motofit_globalreflectometry.ipf

6. GeneticOptimisation.ipf

7. Motofit_loadpackage.ipf

8. Abeles.xop

9. Gencurvefit.xop

10. Multiopenfiles.xop

You will need to open them all for the program to work. The latest version `http://sourceforge.net/projects/motofit/files/latest/download` off the Sourceforge page should place all files in the correct place. It places the files in "~/Documents/Wavemetrics/Igor Pro 6 User Files/" or in "My Documents\Wavemetrics\Igor Pro 6 User Files\". However, if it doesn't you can manually install Motofit.

1. Go to the subversion repository `https://motofit.svn.sourceforge.net/svnroot/motofit/trunk/`.

2. Obtain all the .ipf files listed above.

3. create a directory in "My Documents\Wavemetrics\Igor Pro 6 User Files\User Procedures" or "~/Documents/Wavemetrics/Igor Pro 6 User Files/User Procedures" called *motofit.*

4. Place all the .ipf files in this directory, except the Motofit_loadpackage file. Put this into "My Documents\Wavemetrics\Igor Pro 6 User Files\Igor Procedures".

5. Obtain the .xop files. These pre-compiled DLL's are necessary and make Motofit a lot faster. They can be downloaded from the IgorExchange `http://www.igorexchange.com` website. These normally are placed in the right places via an installer program, but the individual files can be placed by hand into the "My Documents\Wavemetrics\Igor Pro 6 User Files\Igor Extensions" folder.

5   Getting Started

Open all the required procedure files (which should happen automatically if you install using the instructions above). Once the procedures have compiled then a *Motofit* menu should appear on the top menu. Use Fit Reflectivity Data to start fitting your data.

The *Motofit* menu has several sub-options:

1. Fit Reflectivity Data

2. Load Experimental Data

3. Co-refine reflectometry Data

4. SLD calculator

5. Create local Chi2map for requested parameter

6. Fit batch data (this fits multiple datasets, from kinetic data for example).

7. Moto_globalfit

Figure 2: *Motofit* initialisation

5.1 Fit Reflectivity Data

This option creates a user interface for you to fit your data.

When you click on this option a dialogue is brought up. This dialogue enables you to select some of the parameters required to produce the theoretical curve.

Most of the selections are obvious. You can control the Q range for the plot, as well as the number of points in that plot. dq/q relates to the resolution function of your instrument (use dq/q=0 if you want quicker calculations), if you are unsure set this equal to zero. There are 500 points in the SLD profile graph that is created for your layered system. The plot mode controls how the reflectivity is plotted. The mode popup has two options - solvent penetration and complex SLD. The first mode allows you to factor in solvent penetration into each layer. The complex SLD mode enables you to describe each layer with a complex SLD, which is necessary for analysing absorbtion effects.

When you press continue the Reflectivity Panel, Reflectivity graph and Scattering Length Density graph are produced.

The theoretical Q range can be altered at a later stage using the menu item *"Change Q range of theoretical data"*, or from within the reflectivity graph.

There are several sections to the panel:

5.1.1 Model parameters

- NUMBER OF LAYERS : how many layers there are in your model. Layer 1 is closest to the fronting medium, which is where the neutrons are incident, layer n is closest to the backing medium (subphase). You can have as many layers as you like. To increase the number of layers, change this value.

- SCALE : The scale factor for the reflectivity measurement (should be 1, or close to 1). All the values in the theoretical reflectivity curve are multiplied by this number.

The second box allows you to change the parameters for each of the layers.

- THICKNESS : thickness of the each layer (in Angstrom).

- SLD : real part of the scattering length density of each layer.
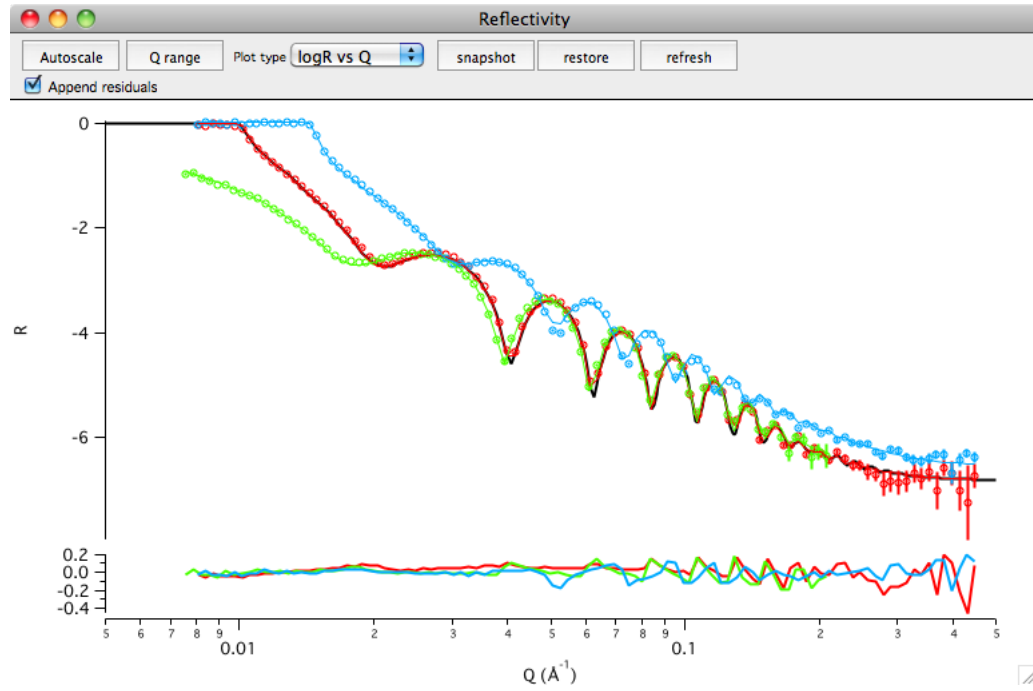
7

Figure 3: The reflectivity graph, showing three different contrasts on a Si/TiO2 film. The residuals are shown at the bottom of the graph.
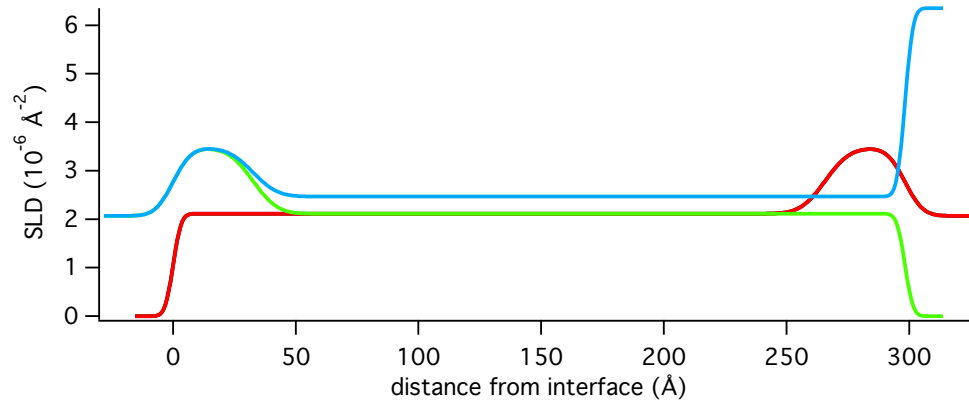

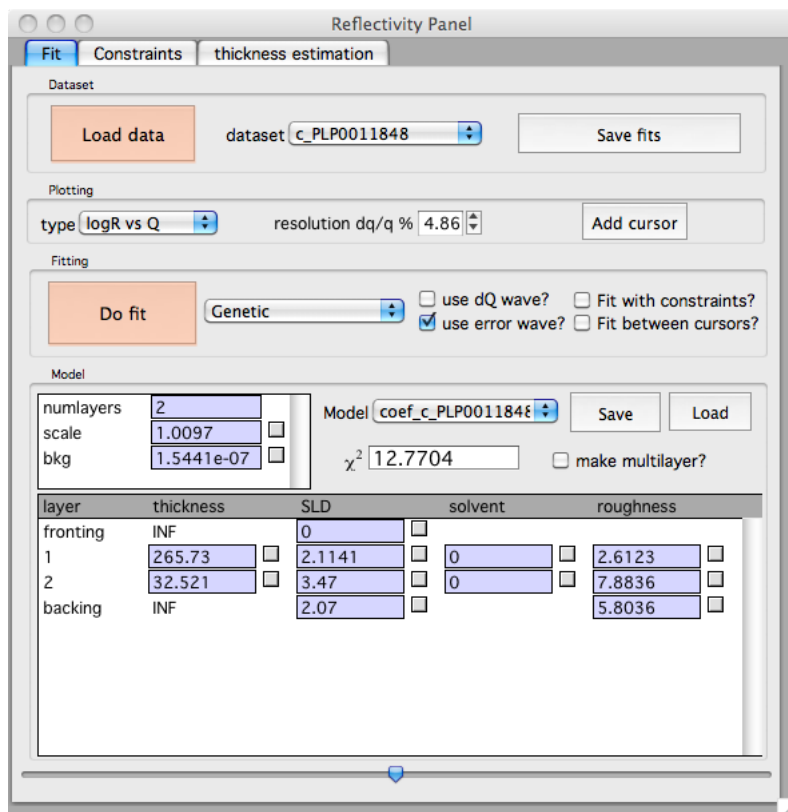Figure 4: The scattering length density graph.

Figure 5: The reflectivity fit panel, setup for a two layer system. Note the presence of a slider at the bottom to vary the last selected parameter
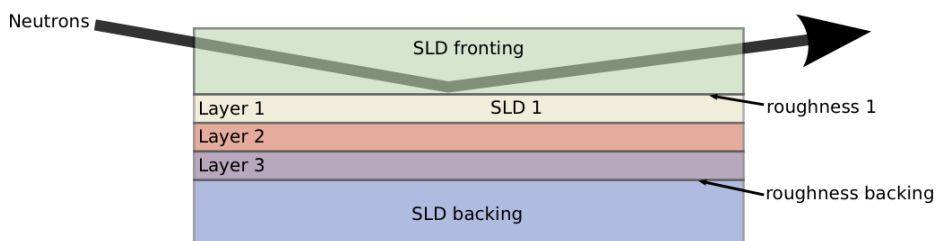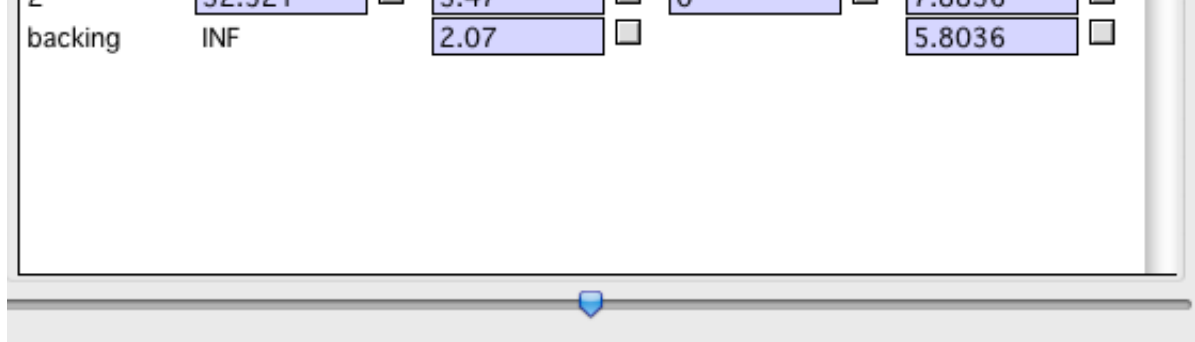

Figure 6: The *Motofit* model

Figure 7: Parameters for each of the layers. The number at the far left hand side of the white box indicates the layer number.

- SOLV : how much of the backing medium (solvent) penetrates into the layer. This is expressed as a percentage. -OR-
- ISLD: the complex part of the scattering length density of each layer

- ROUGH : the roughness at the top of the layer (in Angstrom).

All the real parts of the SLD are expressed as $\rho/10^{-6}\text{Å}^{-2}$. The imaginary part of the SLD is expressed in Å^{-2}. Roughnesses are implemented in terms of an error function, according to Nevot+Croce[4]. In this situation the roughness is the standard deviation of the erf function (Reflfit from NIST uses FWHM for this value, which is approx. 2.35 times larger).

The checkboxes in these tables are used to define which parameters are held during the fit (held=checked).

The model can be saved and loaded by using the SAVE MODEL button. If you want to use a saved parameter file as a starting point then you can use LOAD MODEL, then select the parameter file in the MODEL popup.

The χ^2 value describes how well the current model describes the dataset selected in the DATASET popup. The exact value displayed depends on whether the user has selected the USE ERROR WAVE checkbox and whether a restricted Q range has been selected using the FIT BETWEEN CURSORS? button.

The slider at the bottom of this GUI window is an important new feature of MOTOFIT. If you move the slider from left to right the last selected parameter is continuously varied, with the reflectivity graph and sld graph updating as the parameter changes.

5.1.2 Load data

This button loads experimental data. The program asks if you want to append the data to the theoretical reflectivity curve. When you load data the program removes NaN's and sorts the data as increasing Q_z. You can load either 2, 3 or 4 column data:

Two column data is Q,R

Three column data is Q, R, dR

Four column data is Q, R, dR, dQ.

Q, and its associated uncertainty dQ (FWHM, not standard deviation) are measured in reciprocal Angstroms, Å^{-1}. Only Q and R are required for a fit. The uncertainty (standard deviation) in reflectivity, dR, is required if you wish to weight the data according to counting statistics. dQ is required if you wish to use resolution smearing on a point by point basis.

5.1.3 Dataset / Model Wave

These popup menus select the experimental datasets that you wish to fit. You can save and load coefficient waves using the *save and load model* buttons. This saves a lot of effort.

The coefficient wave is saved as an IGOR text file (*.itx*). This means that you can double click on the file and it loads straight into IGOR, and initialises the reflectivity panel. The saved coefficient wave remembers the exact state of how the data was fitted (including which parameters were held).

5.1.4 Use error wave

If you check this box the fitting process uses the statistical errors on each of the datapoints to weight the fit. See the FAQ section for different weighting scenarios. Please note that your dataset should contain at least 3 columns of data (Q, R, dR).

5.1.5 Plot type

You can change the plot type by simply using this pop up menu. The data fitting operates on the sample plot-type. E.g. if you select RQ4 vs Q, you fit the data as RQ4 vs Q. The plots update as you change the selection in this box.

5.1.6 Resolution

There are two different ways of handling the resolution smearing. The first method assumes that the resolution of the instrument is a constant percentage, e.g. *dq/q=5%*, where dq is defined in 9. This is often the case if the instrument measures at constant footprint.

$$\left(\frac{\Delta Q}{Q}\right)^2 = \left(\frac{\Delta \theta}{\theta}\right)^2 + \left(\frac{\Delta \lambda}{\lambda}\right)^2 \tag{9}$$

In 9 ΔQ is the uncertainty in Q, $\Delta \theta$ is the uncertainty (FWHM) in the incident angle and $\Delta \lambda$ is the uncertainty (FWHM) in the wavelength of the incident radiation. Please note that Motofit assumes that ΔQ is the FWHM of the gaussian, not the standard deviation.

If the data is from a nice instrument that outputs the resolution function as an extra (4th) column (containing dQ values) in the data file, then you can use this to resolution smear the data point by point. This is handy if the resolution varies across the Q range. This is often the case if a single slit setting is used.

The first method is automatically used (as it's faster). To use point by point resolution smearing the select the USE DQ WAVE checkbox.

In both cases smearing is done by gaussian convolution. If you use CONSTANT DQ/Q then the convolution is handled using IGOR's convolve operation and is quite fast. If you select the USE DQ WAVE option then the convolution is performed by the following calculation:

y=f(w,z)
y+=0.1354*f(w,z+2*(dq/2.35482))
y+=0.1354*f(w,z-2*(dq/2.35482))
y+=0.24935*f(w,z-1.666667*(dq/2.35482))
y+=0.24935*f(w,z+1.666667*(dq/2.35482))
y+=0.4111*f(w,z-1.333333*(dq/2.35482))
y+=0.4111*f(w,z+1.333333*(dq/2.35482))
y+=0.60653*f(w,z-1*(dq/2.35482))
y+=0.60653*f(w,z+1*(dq/2.35482))
y+=0.80074*f(w,z-0.6666667*(dq/2.35482))
y+=0.80074*f(w,z+0.6666667*(dq/2.35482))
y+=0.94596*f(w,z-0.3333333*(dq/2.35482))
y+=0.94596*f(w,z+0.3333333*(dq/2.35482))
y *= 0.137023

5.1.7 Fit between cursors

If you want to limit the region of the dataset that the fit function uses then tick this box. Note that you should have both cursors (A and B) on the *dataset* you are fitting.

5.1.8 Use constraints

If you want to use constraints then you should tick this checkbox. Constraints are numbered with respect to their parameter. The parameter numbers are as follows:

SOLVENT PENETRATION

- K0 = number of layers

- K1 = scale factor

- K2 = SLD fronting

- K3 = SLD backing

- K4 = background

- K5 = roughness of backing

- K(4N + 6) = thickness of layer N

- K(4N + 7) = sld of layer N

- K(4N + 8) = solvent penetration in layer N

- K(4N + 9) = roughness between layer N and N-1.

COMPLEX SLD's

- K0 = number of layers

- K1 = scale factor

- K2 = SLD fronting

- K3 = iSLD fronting

- K4 = SLD backing

- K5 = iSLD backing

- K6 = background

- K7 = roughness of backing

- K(4N + 8) = thickness of layer N

- K(4N + 9) = SLD of layer N

- K(4N + 10) = iSLD of layer N

- K(4N + 11) = roughness between layer N and N-1.

For example, the roughness of the first layer is referred to as K9, whilst the thickness of the first layer is K6. The constraints should be entered in the constraints tab, using *add or remove constraint*s You should not leave any blank boxes, otherwise nothing will happen when you try to do the fit. You can either enter simple constraints, e.g. K6>5 or complicated constraints (K10>5*K2). For further details on valid constraints please refer to the Wavemetrics manual.

NOTE: If you have ticked the use constraints box, then you must have at least one constraint, or nothing will appear to happen. In a similar manner, nothing will happen if you enter a constraint then don't allow it to vary when you do the fit.

CONSTRAINTS DON'T APPLY DURING GENETIC OPTIMISATION!

5.1.9   Make multilayer

This option is in the process of being implemented for the latest version of motofit. If you need multilayer systems then please use Motofit 3.

5.1.10   Do fit

Fits the experimental dataset with the theoretical model. You have a choice of using GENETIC OPTIMISATION, LEVENBERG-MARQUARDT, GENETIC OPTIMISATION FOLLOWED BY LEVENBERG MARQUARDT or GENETIC OPTIMISATION And MONTE CARLO ERROR ANALYSIS as the fitting algorithms. Each of these methods minimise the χ^2 value.

Output available from the fit   All output data from the fit is placed into the datafolder for that dataset. Please see 5.4.1.

1. Fit curves for the data you analysed. An SLD profile is also appended to the SCATTERING LENGTH DENSITY GRAPH.

2. residuals for the data you analysed (difference between theoretical model and actual data).

3. a coefficient wave containing the parameters, called COEF_DATASETNAME_R.

4. a wave, W_SIGMA, containing the uncertainty (1 SD) in each of the fit parameters.

5. The covariance matrix is contained in a wave called M_COVAR.

6. the fit history for all the fits performed on that dataset is contained in a wave called FIT_HISTORY.

7. An IGOR notebook called 'REFLECTIVITY FITTING REPORT' is produced. This notebook contains a record of all the reflectometry fits that you perform in an experiment (containing information on fit parameters, fit errors, a graph of the fitted data, the χ^2value, time, date, and the method of fit).

8. A wave called FIT_HISTORY is created/appended to. This wave contains all the fits done on this dataset (so you can retrieve them if things go wrong).

9. The history window contains the fitted parameters, associated uncertainties (+/- 1 standard deviation) and best χ^2 value for the fit.

5.1.11   Genetic Optimisation

This method is very efficient at finding global minima in the Chi2 function, even if the starting guesses are far from the final solution. The algorithm used in the program is a differential evolution technique from Wormington et al[7] and Storn and Price[6]. This method should be the first point of call if confronted with a difficult reflectivity pattern. I suggest using this as the default fitting procedure.

The program asks for information to set up the genetic optimisation.

- *Number of generations.* This is effectively the number of iterations you want to do. The more generations you use, the longer it takes, but you should expect a closer answer. I suggest trying 100 generations to start with.

- *Mutation constant.* The mutation constant is used to create genetic diversity. A larger number means more genetic diversity, but may mean that the solution takes longer to find. This value should lie between 0 and 1. Start out with 0.7 and see how you go.
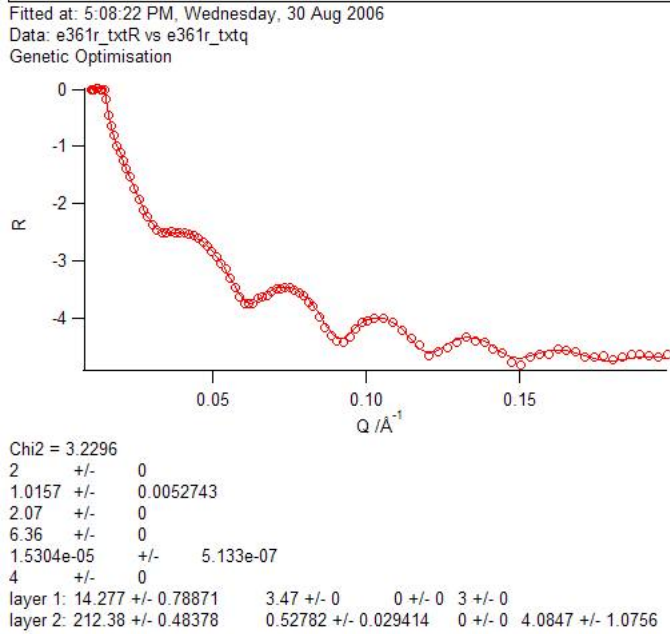
Figure 8: A report is produced for each fit, so you can compare different approaches.

- *Population size multiplier.* The genetic algorithm works by starting off with a large population of initial randomised solutions (that lie in between the limits you set), which are then refined. The size of this population is set by *popsizemultiplier*the number of free parameters.* If you have a large parameter space to examine, then it may help to set this value larger, but 10 has worked in the past.

- *Recombination constant.* Should lie between 0 and 1. This controls how much of the original guess is used in the evolved guess. If you use a low number then the original guess will not evolve that much. If you use a large number then the guess will evolve a lot, making it more likely to find a solution, but taking longer in the process. Try starting out with 0.5.

- *Fractional tolerance to terminate fit.* If the standard deviation of all the population χ^2 values divided by the mean χ^2 value is less than this number, then the fit will terminate. 0.05 is a typical value that should be used.

The user is then presented with a table showing the initial guesses, the limits to be placed on those parameters during the fit. The optimisation is unable to search for solutions outside those bounds. When you are satisfied with the setup press *continue* and the optimisation will start. If the limits are not set correctly (e.g. your upper limit is less than your lower limit) you will be asked to set them up again.

On finishing the optimisation (or aborting it part way through) a coefficient wave and fitwaves will be produced, a pointwise estimate of the parameter uncertainties is contained in the wave *W_Sigma*.

NOTE: CONSTRAINTS DO NOT APPLY DURING GENETIC OPTIMISATION.

5.1.12 Levenberg-Marquardt

The Levenberg-Marquardt (LM) method is a much faster and efficient method for fitting than genetic optimisation. However, it only has a good chance of working if your initial guess is close to the final fit, as it has problems escaping from local minima in the χ^2 function. It is

often best to use genetic optimisation to find the best solution, then employ LM to speedily take you the rest of the way. LM has the advantage of being able to employ inter-parameter constraints, whereas genetic optimisation is only able to set boundary limits on parameters.

5.1.13 Genetic - LM

This option performs a genetic optimisation followed by a Levenberg-Marquardt fit, the best of both worlds.

NOTE:

1. Remember to check the final answer, to make sure the model is something that is physically realistic.

2. Negative numbers may appear as fit results, for the thickness and roughness. DON'T PANIC. Internally in *Motofit* the absolute value of these numbers are taken, i.e. abs(thickness).

5.1.14 Genetic + Monte Carlo Error analysis

Normally, when one uses the Levenberg Marquardt or Genetic Optimisation methods the parameter uncertainties are worked out from the covariance matrix6.12. This assumes that the fitted parameter probability distribution is normally distributed (the mean being the fitted value and the s.d. the parameter uncertainty). Sometimes this assumption is incorrect. Alternatively, the technique used to obtain the covariance matrix may be unstable - the most used method requires the inversion of the Hessian matrix (the partial derivatives of χ^2 with respect to each parameter). If the fit is insensitive to some parameters, or the parameters are radically different in size, then the inversion may be numerically unstable and the parameter uncertainties can be estimated incorrectly.

The best way of estimating the probability distribution of the fitted parameters is to use a Monte Carlo error analysis[2]. In this method the data is refitted a large number of times (e.g. $M = 1000$). Each time the data is fitted a new dataset is synthesised by adding a random gaussian deviate onto each datapoint. The random gaussian deviate is taken from a distribution which has a standard deviation equal to the measurement error on that datapoint (thus, if one averaged the M datasets one would obtain the exact same dataset). The data is fitted with genetic optimisation. When the fit finishes you are left with M parameter values. The mean of those M values is the fitted parameter value and the standard deviation of the M values is the parameter uncertainty. MOTOFIT automatically calculates these values, as well as working out the covariance matrix for all the parameters, as well as a plot showing the variation in scattering length density profiles9.

Please note that this fit option can take a fairly long time. There are ways of speeding it up, I have developed cluster programs that can reduce the fitting time by several orders of magnitude. Please contact me for further details.

5.1.15 FFT of data / Fringe spacing

The Fourier transform of the selected dataset (from the dataset popup) is displayed in the third tab of the reflectivity panel. If you have oscillations in your reflectivity curve you will get a peak in the fourier transform that corresponds to your layer thickness. Use the cursor in the Fourier transform plot to get values for each of the peak positions. This option is excellent for setting up fits.

If you have many layers present then you will get multiple peaks present in the fourier transform. You can change the region over which the FFT is applied. The low Q region should be above the critical edge of the data, and the high Q end should stop before the background is reached.
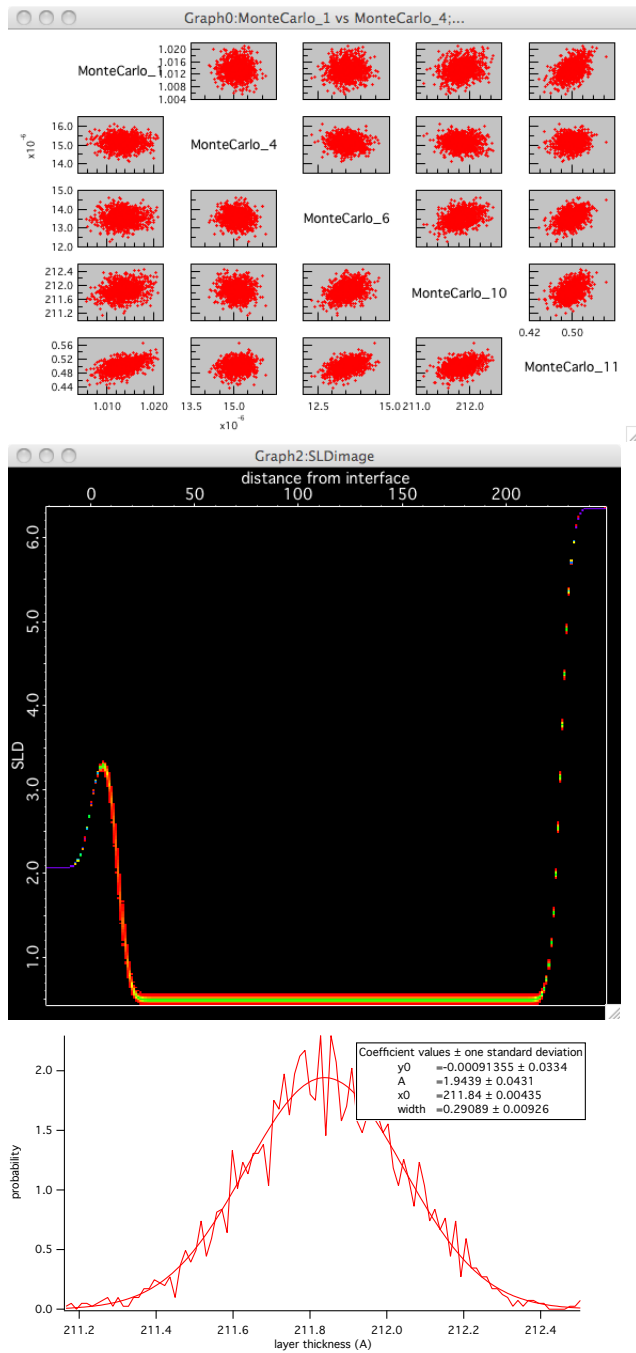
Figure 9: Output from Monte Carlo error analysis. First graph shows the covariance between parameters. Second graph shows the family of likeliest SLD profiles. Third graph shows the histogrammed thickness parameter for the layer of interest.
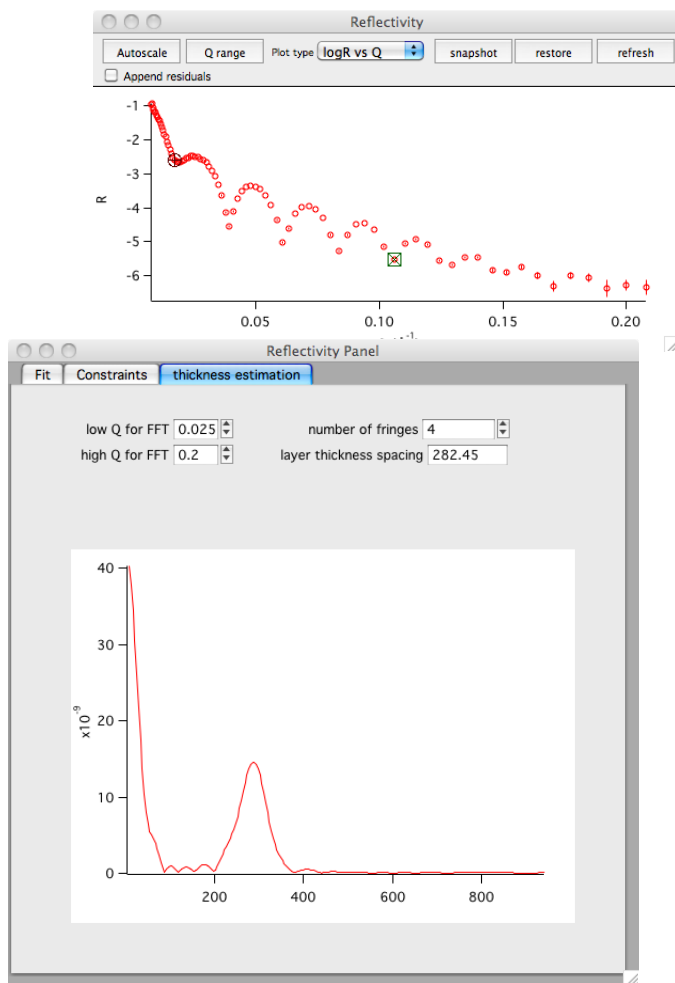
Figure 10: Fourier transform of some neutron data. This data fitted to a 350A layer

(for the techno-nerd: the reflectivity curve is multiplied by Q4, then Fourier transformed with an FFT. The data is zero padded before the fourier transform, and a cosine window is applied during the transform).

If you have multiple Kiessig fringes then you can estimate layer thicknesses by assessing the *fringe spacing*.

Put the cursors on the reflectivity graph by pressing CURSORS. This button appends the cursors to the trace in the DATASET popup. Move the cursors until you have selected an integer number of fringes, change the number of fringes, the layer thickness is then displayed.

5.1.16 χ^2 value and other statistics

As the model is updated the χ^2 value is also updated, which tells you if your fit is improving. The χ^2 value is normalised to the number of points.

$$\chi^2 = \sum_{n=1}^{L} \frac{1}{L-P} \left(\frac{y_{n,obs} - y_{n,calc}}{\sigma_{n,error}} \right)^2 \tag{10}$$

If the USE ERROR WAVE is not ticked $\sigma_{n,error}$ is set to 1 for all points, if the checkbox is selected then the experimentally determined measurement errors are used to fit the data. After fitting data the normalised correlation matrix (M_Covar) is produced in the datafolder for that dataset, which indicates interdependence of fit parameters. When fitting the data this value should be as low as possible, if your model is correct and you have weighted the data then the optimal value of χ^2 is 1. This means the average squared deviation from the datapoints is similar to the variance of the measured error.

5.2 Reflectivity Graph

This plot shows the theoretical reflectivity curve (3) for the layer model specified in the REFLECTIVITY PANEL, as well as all the datasets. Whenever you change a parameter in the panel then the graph will update. The main part of the graph is the theoretical reflectivity profile.

Q RANGE - changes the theoretical Q range of the plot

PLOT TYPE - changes the plot type.

SNAPHOT - creates a snapshot of the theoretical model, allowing you to compare different scenarios.

RESTORE - restore the state of the theoretical model before a fit was started.

REFRESH - reloads the data from file (useful if you are still acquiring the data.

APPEND RESIDUALS - the difference between the fit and the data is shown.

If you have the full version of IGOR you can export the graph in whatever way you would like. This manual was created with the EPS export.

5.3 Scattering Length Density Graph

This plot shows the SLD profile corresponding to the theoretical model, as well as the SLD profiles for each of the fitted datasets.

The SLD profile is calculated using 11. Where N is the total number of layers, z is the distance from the top interface and *erf* is the error function.

$$\rho_z = \sum_{i=0}^{N} \frac{\rho_i - \rho_{i+1}}{2} \left(1 + \text{erf} \left(\frac{z - z_i}{\sqrt{2}\sigma_i} \right) \right) \tag{11}$$

If you have the full version of IGOR you can export the graph in whatever way you would like. This manual was created with the EPS export.

5.4 Load Reflectivity data

When you bring up this option from the top menu you load in 2, 3 or 4 column reflectivity data. The data format is tab delimited, with the data columns as:

Q, R, dR, dQ

Q is in units of $^{-1}$, R is the reflectivity, dR is the measurement error for the corresponding reflectivity value (standard deviation) and dQ is the uncertainty in Q value (FWHM).

If you have a graph already open (e.g. you have already opened the reflectivity panel), and this graph is on top, then it asks if you want to append it to that graph, which is quite useful. In other words, you see your experimental data and the theoretical data on the same plot. If there is no graph already open, then it creates a new graph. You can then open (and append) further reflectivity datasets to this graph.

5.4.1 The organisation of loaded data in IGOR

When data is loaded into IGOR a separate data folder is created for each dataset. If your dataset is called MYDATASET, the data will reside in the datafolder:

    root:data:mydataset

These datafolders contain the Q/R/dR/dQ data, and will also contain the fitted parameters, and fitted uncertainties after the fit has finished. You can investigate the data folder structure of IGOR by bringing up the databrowser with the DATA->DATABROWSER menu items.

5.5 Fitting Multiple Contrasts / Global Fitting

Fitting multiple (neutron + X-ray) contrasts is a cinch in *Motofit*. You use the global fit package. Simply follow the following steps.

1. Load all the experimental data (preferably in the same graph).

2. Bring up the co-refinement panel MOTOFIT-> CO-REFINE REFLECTOMETRY DATA.

3. Select which datasets you would like to fit using the ADD DATASET and REMOVE DATASET buttons.

4. Link the parameters which are common. You can have any linkage you like. Select the parameters you wish to join using a multiple selection, then press LINK SELECTION.

5. Select the COEFFICIENTS tab. Enter all required parameters. Please note that you can right click on a cell entry, this brings up a dialogue where you can fill out all the cells with a coefficient wave that has already been used to fit the dataset, e.g. from an earlier analysis. The number of layers NEEDS to be correct.

6. Use the slider at the bottom of the COEFFICIENTS tab to continuously vary a parameter. The reflectivity and SLD graphs continuously update as parameters in this tables are changed.

7. Press FIT. The fit proceeds using the options currently selected in the REFLECTVITY PANEL, such as USE DQ WAVE, USE ERRORS or the fit method (Genetic/Levenberg Marquardt, etc).

8. The output from the fits are placed in the datafolder for that dataset.

9. You can fit as many datasets (Xray or Neutron) as you wish, but be careful when co-refinining X-ray and Neutron data at the same time.

5.6 Scattering Length Density Database

As Motofit v3.04 a scattering length density calculator is integrated into the program, along with the SLD database. These features can be accessed from the reflectivity menu, or from the SLDdatabase panel created when Motofit is initialised.

Figure 12 demonstrates a scattering length density calculation. One enters the chemical formula (including isotope information) and density and the Neutron and X-ray scattering lengths (8047eV) are calculated for you. The typing of the chemical formula is strict and case sensitive. One can use fractional atomic compositions

Formulas that work:

- H(2)2O - heavy water (this is also the same as D2O)

- D2O

- Al2O3

- AlO

- Al1O1

- Al2.023O3.056 (fraction compositions allowed)

Formulas that don't work

- h(2)2o (Elements are case sensitive)

- al2O3 (Elements are case sensitive)

- H()2O (there is no isotope information)

- H(100)2O (there is no isotope of Hydrogen with weight of 100).

The SLD calculation is performed using the method described in Section **??**. The calculator uses the neutron lengths tabulated at the NIST `http://www.ncnr.nist.gov/resources/n-lengths/` website. The X-ray scattering lengths were obtained from the scattering factors on the Lawrence Livermore `http://www-phys.llnl.gov/Research/scattering/asf.html` website.

You can then enter the calculated value into the SLD database, on the next tab.

If you can't remember all the SLD values that you use then you can add them to the database. You can add as many new values as you like. (You can edit the database using Excel as well). The SLD database is automatically loaded when the reflectivity macro's are initialised, so should be present in the *Motofit* folder. Values from the SLD database can be automatically inserted into a structural model by right clicking in the SLD cells in the reflectivity panel (section **??**). The format for new lines of the SLDdatabase is:

Chemname <tab> chemformula <tab> SLDneut <tab> SLDXray <tab> massdensity.

The last tab of the SLDdatabase panel enables the user to calculate SLD's from the ideal mixing of two components.

5.7 Batch fitting datasets

This function is essential for people with many kinetic datasets. When you select *Load batch data* (from the *Reflectivity* menu) you are prompted for a folder in which your data resides. The program then loads in all the files in that directory, with the extension you provide. Please take care, if you specify loading all ".txt" files, then move any non-reflectivity text files from that directory beforehand. You can also specify an offset for the plotting, this offsets all the datasets in the batchdata graph so that you can see each of them clearly. The data is placed in the MOTOFIT_BATCHFITS datafolder.

Now use the reflectivity panel to set up the parameters for the first dataset (you should try fitting that dataset before loading the batch data). You can fit with Genetic Optimisation or Levenberg Marquardt (or both). All options, such as USE ERROR WAVE and FIT BETWEEN CURSORS, from the reflectivity panel are used for each of the batch fits.

Now select FIT BATCH DATA from the menu MOTOFIT->BATCHDATA. This fits the first dataset (as normal), using the parameters in the reflectivity panel as a starting point. It then uses the fit parameters from that first dataset as a starting point for the second dataset. The program then cycles through all the datasets fitting them all. In testing around 100 files were fitted in 5 minutes. All the fit curves, coefficients and errors are collected as the batch fit proceeds. Comprehensive graphs detailing the fits and trends in parameters are available at the end, ready to export to your Nature paper. Please bear in mind that this technique cannot handle with a system that changes in structure, i.e. you transition from a single layer model to a two layer model. Also, if you use Genetic Optimisation the limits that you set must be suitable for all datasets.

5.8   Create local chi2map for requested parameter

This option is available from the *Motofit* menu. You can use this to explore how Chi2 varies for selected parameters, around their current values.

If you choose the 1D version you are prompted to enter a parameter number (available from the reflectivity panel), and a range to explore (expressed as a percentage of the value). A graph is then displayed of Chi2 versus parameter value.

The 2D version allows you to examine a χ^2 surface, as a function of two parameters. For example you can look at how Chi2 varies as a function of layer thickness and layer scattering length density. The following figure demonstrates that χ^2 is sensitive to the layer thickness, but not the SLD.

5.9   Fitting user defined functions SLD profiles/analytical profiles.

If you have a specific profile to fit, then this will probably replace MOTOFIT. In otherwords, you have a certain SLD profile you want to create, with its own set of parameters. This may take the form of e.g. gaussian distribution of a grafted polymer, tanh roughness between interdiffusing polymer layers, chebyshev polynomials describing a free form SLD profile, fitting area per molecule to a lipid system, etc.

In this last example parameterisation using an area-per-molecule reduces the number of parameters by one. Normally one fits two thicknesses and two SLD values for a lipid layer (headgroup and tailgroup). However, the volume of the head and tail groups, and their scattering lengths are known. Thus, one can fit the data using an area-per-molecule and head and tail group thicknesses - 3 parameters instead of 4. This kind of parameterisation is more stable because it automatically enforces the one head group per tail group constraint. In addition, it saves an additional calculation working out the area-per-molecule.

In this situation you will have to use the curve-fitting dialogue (ANALYSIS->CURVE FITTING or ANALYSIS -> GENETIC CURVEFITTING). You will create a specific parameter wave for your model. You will then write a function that will take your parameter wave as input, create a specific wave (see 5.1.8) which is used by the kernel that calculates the reflectivity, then calculate the reflectivity. Please contact me `mailto:andrew_nelson@users.sourceforge.net` for the code required to do this. This is very simple, so don't be afraid to ask.

5.10   Test datasets

The distribution package contains several test datasets (can you fit them?). Also included are the coefficient waves for each dataset. The resolution for these experiments is dQ/Q = 4%

Dataset e179 is a multilayer sample. This sample consists of a two layer stack of Ti/Ni, which has 25 repeats. The multilayer is appended to layer 0.

e361r, e365r and e366r are solid-liquid datasets of silicon vs water. On top of the silicon substrate is a native SiO_2 layer, on top of that is a polymer layer; which makes it a two layer system. The only difference between the different datasets is the scattering length density of the water phase. Of the 14 parameters (0 –> 13 inclusive) the only parameters that aren't global are: *SLDbase, bkg, SLD* (layer 2). Therefore these datasets are ideal for the global fit routines.

Sitest.txt is a solid air experiment of a silicon wafer vs air. The Si wafer has a native oxide layer (SiO_2) of around 15A thickness on top.

c_PLP0011848.xml, c_PLP0011854.xml and c_PLP0011859.xml are several measurements on a TiO2 system. 11848 is air/TiO2/SiO2/Si, 11854 is Si/SiO2/TiO2/air and 11859 is Si/SiO2/TiO2/D2O.

6 Frequently asked questions

6.1 How do I calculate a scattering length density?

The easiest way is to use the inbuilt SLD calculator.

But you should also be able to calculate them for yourself. The formula for calculating a scattering length density is given in 12. In this equation V_m is the molecular volume of the material and b_{c_i} is the bound coherent scattering length of the i th atom of a molecule with n atoms.

$$\rho = \frac{\sum_{i=1}^{n} b_{c_i}}{V_m} \tag{12}$$

You can also write this as 13, where N_a is Avogadro's number, ρ_{mass} is the mass density of the material and M_R is its relative molecular mass.

$$\rho = \frac{N_a \rho_{mass}}{M_R} \times \sum_{i=1}^{n} b_{c_i} \tag{13}$$

You can find a table of neutron scattering lengths at:

- http://www.ncnr.nist.gov/resources/n-lengths/

For X-rays you have to calculate the scattering length for each atom before doing the calculation, 14.

$$b_i = \frac{e^2}{4\pi\epsilon_0 mc^2} f_{1i} \tag{14}$$

where e is the charge on a single electron, ϵ_0 is the permittivity of free space, m is the mass of an electron and c is the speed of light. f_{1i} is the scattering factor for an atom of element i and may be found from:

- http://www-phys.llnl.gov/Research/scattering/asf.html

To use this website you will need to know the wavelength of the X-ray radiation that you are using (1.54Å=8047eV).

This is an exact way of calculating b_i, a much quicker approximation is to use 15, where Z is the atomic number and r_e is the Compton radius ($r_e = 2.8179 \times 10^{-5}$Å)

$$b_i = Z r_e \tag{15}$$

To find out how to enter the scattering length densities are entered into the program please see Section 5.1.1.

6.2 Why can't I install the program properly?

Did you follow the installation instructions carefully? If you think you have:

- make sure that you have loaded all the required files.

- update to the latest version of IGOR.

- if all these steps fail then please email:
  mailto:Andrew_Nelson@users.sourceforge.net

6.3 What is a wave in IGOR?

A wave is a data object, which contains a column of data, similar to Excel. When you load a reflectivity dataset with Q, R and dR, you are loading 3 waves. One wave contains the Q values, and there is a wave for the Reflectivity values and a wave for the error in reflectivity values. Obviously Q,R and dR should have the same number of points. Each of the waves has a different name. If you opened the same dataset in Excel you would see 3 columns containing the Q,R and dR values. IGOR is a program that performs mathematical operations on these waves.

6.4 What is a coefficient wave?

A coefficient wave is a wave containing the parameters for the fit. They can be saved as IGOR text files (*.itx), which have some extra advantages to normal text files, try double clicking on one of the coefficient files.

6.5 What should I call my coefficient waves when I save them to disk?

You can save them as anything, but call it something you will remember.

6.6 Why are the layer thicknesses and roughnesses sometimes negative after the fit?

Motofit uses the modulus of the thickness and roughness values, so there is no need to worry.

6.7 The model doesn't seem to be sensitive to roughness?

Sometimes you may fit some data and some of the roughness parameters appear to go haywire. This is because in a lot of situations the modelled data isn't sensitive to roughness and when you start to fit the roughnesses oscillate dramatically. This often occurs in situations where there aren't many features in the reflectivity profile and where the data doesn't extend to large Q values. To combat this change the troublesome roughnesses to a realistic value, then fix them during the fit. Once you get close to a good solution then this problem usually gets better. Be sure to look at the correlation matrix to determine any problems.

6.8 Can I fit X-ray data and Neutron data at the same time?

Using global fit you can fit as many neutron contrasts as you like. You can also fit X-ray and neutron data together. However, difficulties may arise because X-ray datasets normally have much better statistics than neutron data. Therefore when you try to fit neutrons and X-rays together the data is sometimes preferentially weighted towards the X-ray set.

6.9 What format should my dataset be in?

The dataset should have either 2,3 or 4 columns. If there is a two column dataset then the columns should consist of Q and R values. For a three column dataset you should have Q, R and dR (where dR is error in reflectivity). For a four column dataset you should have Q, R, dR and dQ (where dQ is uncertainty in Q). The reflectivity values should be a value between 0 and 1. i.e. you cant load in Q, logR, dR or Q, RQ^4, dR data. You should get rid of errant datapoints which have R<0 (which may happen if you subtract a background). All columns should be the same length. For best results the file should be saved without a header (i.e. the data starts in the first line), and in the tab delimited format. This file manipulation is easy in Excel.

6.10 When should I use Levenberg-Marquardt or genetic optimisation?

Levenberg Marquardt is fast, but can get stuck in local minima - solutions which aren't the best - especially if the starting solutions are not great. Genetic Optimisation has a higher probability of finding the correct solution, even if the starting solution is not good, but is slower.

6.11 How are the parameter uncertainties calculated?

See 6.12. Both Levenberg Marquardt and the Genetic Optimisation approach use a finite difference approach to estimate parameter uncertainty. For a more robust (but much slower) approach see 5.1.14.

6.12 What is a correlation/covariance matrix?

The covariance matrix is the inverse of the Hessian matrix, which describes the concavity of χ^2 with respect to each fit parameter (the second order partial derivative of χ^2 with respect to the parameters). The leading diagonal of the covariance matrix gives you the *variance* (remember, standard deviation is the square root of the variance) in uncertainty of the fitted parameter, whereas the off diagonal elements tell you about the *covariance* (interdependence) of parameters.

There should be an inverse relationship between the second-order derivative (of χ^2) for a parameter and its variance. If the minimum of the χ^2 function is sharp with respect to a parameter (see 15, then the change of slope of χ^2 around the minimum is sharp and the second order derivative is large. In such cases the parameter estimate is stable and the variance in parameter estimate is low. However, if the χ^2 function is not sharp, because the parameter can move greatly without affecting the χ^2 value, then the change in slope around the minimum is negligible and the second order derivative is nearly zero. As a result the variance in parameter estimate is large.

The Hessian matrix is calculated using a finite difference approximation, then inverted to give the covariance matrix.

In some cases the matrix inversion is unstable and may give incorrect values for the variance/standard deviation of the parameter estimate. In such cases use the Monte Carlo resampling technique (5.1.14) for a robust estimation of the parameter standard deviations.

When the covariance matrix is normalised (to give you a correlation matrix) the off diagonal elements will have values in the range -1 to 1. A value of 1 means perfect correlation, i.e. as one parameter changes the other parameter will increase concomitantly. A value of -1 means perfect inverse-correlation, as one parameter increases the other will decrease concomitantly. The optimal value should be 0, but values less than ~0.7 are ok. Too large a value means that there is over parameterisation in your model. For example, if you fit a two layer model to a dataset that really only requires a single layer you may find that the

layer thicknesses for each of the layers have a high degree of inverse-correlation - they vary to keep the overall layer thickness constant.

6.13 I want a covariance matrix, how do I get it?

The covariance matrix is automatically produced after a fit. It is stored in the datafolder for that dataset:

```
root:data:mydataset:M_covar
```

To get this displayed you will have to use the databrowser.

6.14 What is the databrowser?

The databrowser is an IOOR method for displaying how the data is organised in IGOR (similar to a filesystem). To bring up the databrowser select the menu items DATA–>DATABROWSER. Each dataset is kept in a separate datafolder: ROOT:DATA:MYDATASET:

6.15 What type of weighting should I use?

Here we explore the χ^2 function for different types of weighting. I have plotted χ^2 as a function of Q for all the types I use and they are listed below. I have also investigated the situation where dR=1 (i.e. weighting has no effect). This should give a guide on how to weight data. Note that the investigation is on a single dataset only. In addition, the SLD of the top and subphase is fixed. This means that if I'm slightly off with those values then there might be spikes in χ^2 at low Q. I created all these graphs sequentially, from situation a). In other words I performed a) first, then used this as a starting point for all the other fits. This is a significant assumption when drawing conclusions, as this does not take into account the fact that each of the different ways of weighting may have different 'descent pathways' to the global minimum. Thus one should consider various starting points.

- fitting as log R vs Q, no weighting (dR=1). As you see this option has a good fit, and the χ^2 function starts to weight the data towards high Q (which in this case isn't bad).
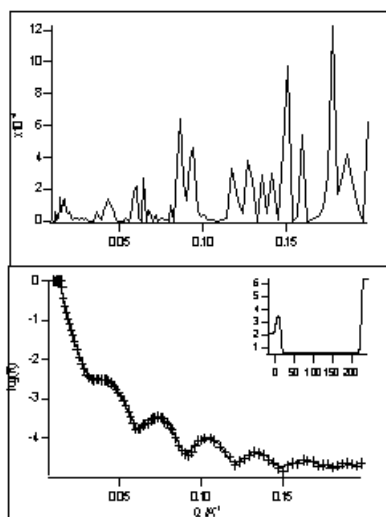

- Fitting logR vs Q, with errors. When you take this option you get rid of the bias towards the higher Q end, and the program will essentially fit uniformly over the Q

range. However, you have to be careful with low Q data, as poor fitting around the critical edge (sometimes not unusual) will increase χ^2 significantly and may weight towards low Q, which is bad!
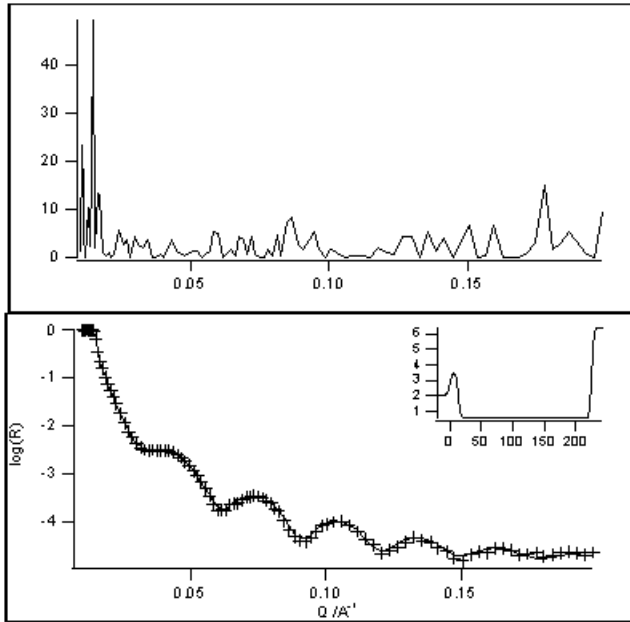

- Fit R vs Q, no weighting This one really doesn't work, never try this option. This is appalling.
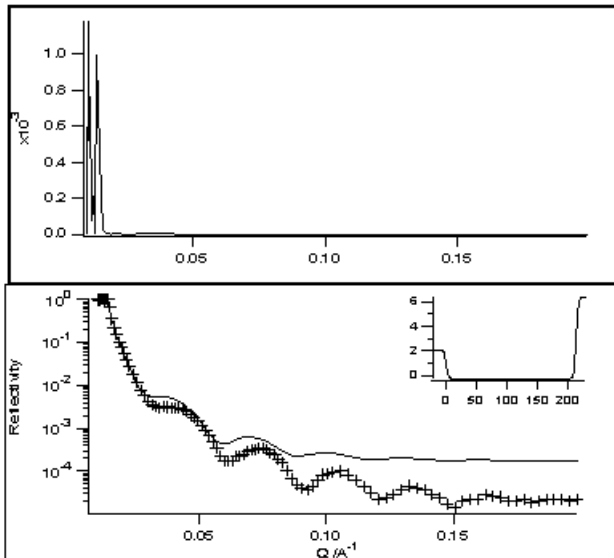
- Fitting R vs Q, with error weighting. Ok, but similar to b). In this situation you really need to make sure that the fit is good around the critical edge, or the fit is bad. The fit is often bad around the critical edge if you are not handling the resolution correctly, or if you don't allow the top or subphase SLD's to vary (which control the position of the critical edge).
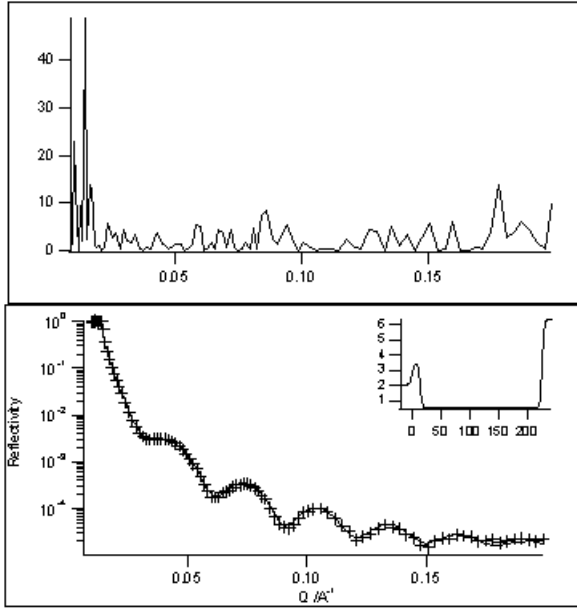
- Fitting R vs Q, weighting as dR=1/R. This is often used as a way of weighting, as it handles the entire Q range successfully. I like this option. Unfortunately you're not using errors.
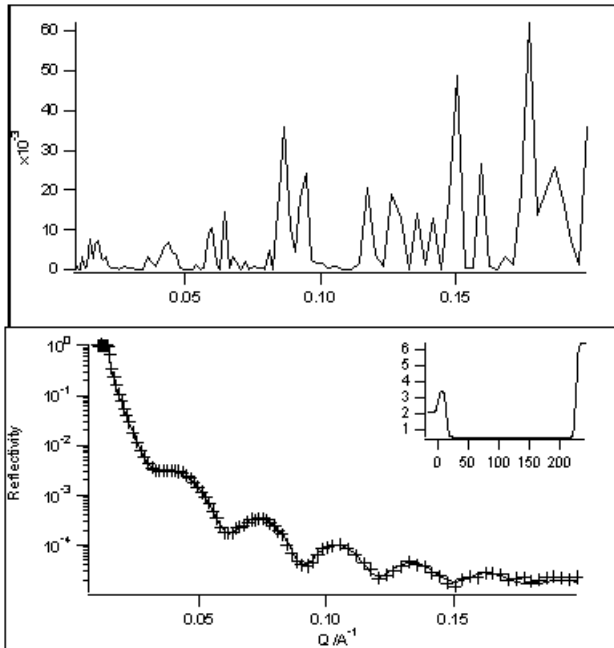
- Fitting as RQ^4, no weighting. Seems to handle entire Q range ok, but there is some bias towards high Q.
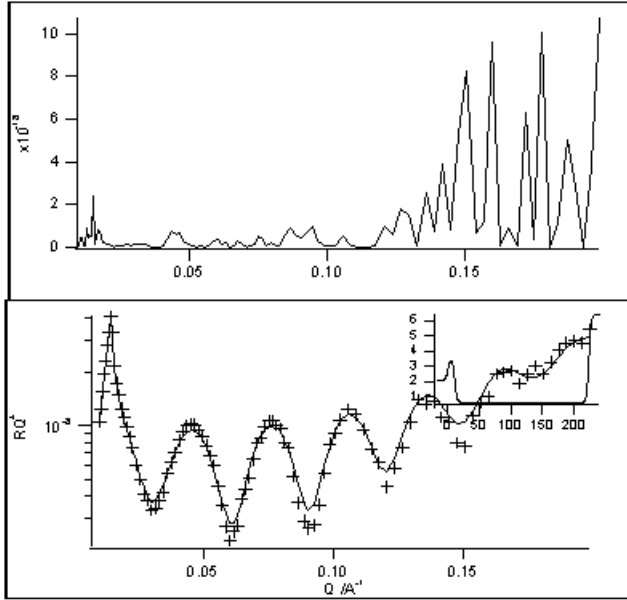
- Fitting as RQ^4 vs Q, with errors. This one is telling us that the fit is correctly weighting all the Q range, but will show some bias towards low Q, if the critical edge isn't handled correctly.
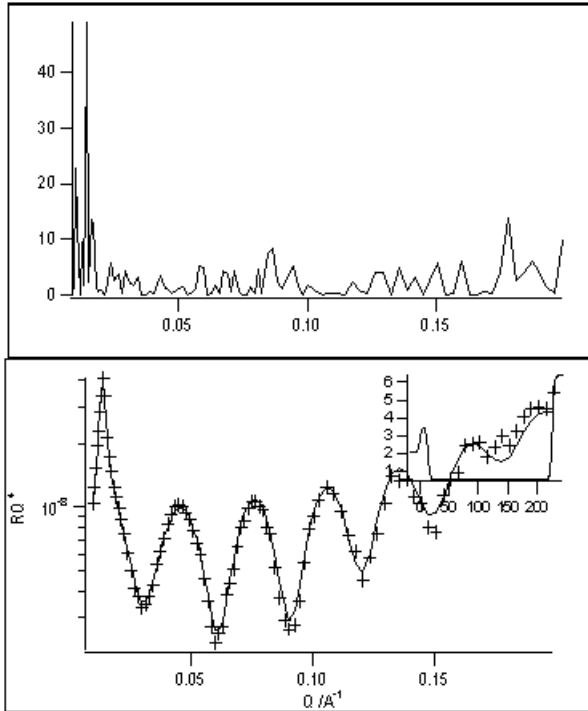
So what are the conclusions? These examples show that it's essential to make sure the critical edge is handled correctly if you want to fit with the error bars! This is because all the examples with error weighting have spikes in the χ^2 function at low Q. Of course, the critical edge region is the part of the reflectivity curve that has the least information about the sample. If I was fitting with error bars I would do these types of plots to make sure that

the crit edge region was being handled correctly, otherwise your fits may be skewed towards low Q. Fitting programs and reduction programs should make sure that the resolution is being used properly, in order to remove this problem.

The graphs show that fitting without error bars is quite robust if you fit as logR vs Q or RQ^4 vs Q (R vs Q doesn't work). This is fairly effective at weighting equally over the entire Q range, but there is a slight risk of bias towards higher Q.

Overall I would use all the options except c). The graphs show that there isn't really an advantage to fitting one way or another, so I suggest that it's a fallacy to say that fitting as RQ^4 vs Q is better! All this option offers is the removal of the Fresnel influence, which makes it easier to see what is happening to all the layers. Of course this is a simple analysis, so take these conclusions with a pinch of salt.

7 Bugs

Send me details of any bugs and I'll fix them. I have benchmarked the program against other programs and I know that the reflectivity is calculated correctly. If you have any suggestions on how to improve things/would like to help develop things then please email me.
`mailto:Andrew_Nelson@users.sourceforge.net`

References

[1] OS Heavens. *Optical Properties of Thin Films*. Butterworth, London, 1955. Optical Properties of Thin Films.

[2] F. Heinrich, T. Ng, D.J. Vanderah, S. Prabhanshu, M. Mihailescu, and M. Losche. A new lipid anchor for sparsely tethered bilayer lipid membranes. *Langmuir*, 25:4219–4229, 2009.

[3] A. Nelson. Co-refinement of multiple contrast neutron / x-ray reflectivity data using motofit. *Journal of Applied Crystallography*, 39:273–276, 2006. Co-refinement of multiple contrast neutron / X-ray reflectivity data using MOTOFIT.

[4] L. Nevot and P. Croce. Caracterisation des surfaces par reflexion rasante de rayons x. application a l'etude du polissage de quelques verres silicates. *Revue de physique appliquee*, 15:761–769, 1980. Caracterisation des surfaces par reflexion rasante de rayons X. Application a l'etude du polissage de quelques verres silicates.

[5] L.G. Parratt. Surface studies of solids by total reflection of x-rays. *Physical Review*, 95(2):359–369, 1954.

[6] R Storn and K.V. Price. Differential evolution - a simple and efficient scheme for global optimization over continuous spaces. Technical report, -1, 1995. ftp://ftp.icsi.berkeley.edu/pub/techreports/1995/tr-95-012.ps.gz.

[7] M. Wormington, C. Panaccione, K.M. Matney, and D.K. Bowen. Characterization of structures from x-ray scattering data using genetic algorithms. *Philosophical Transactions of the Royal Society of London*, 357:2827–2848, 1999. Characterization of structures from X-ray scattering data using genetic algorithms.
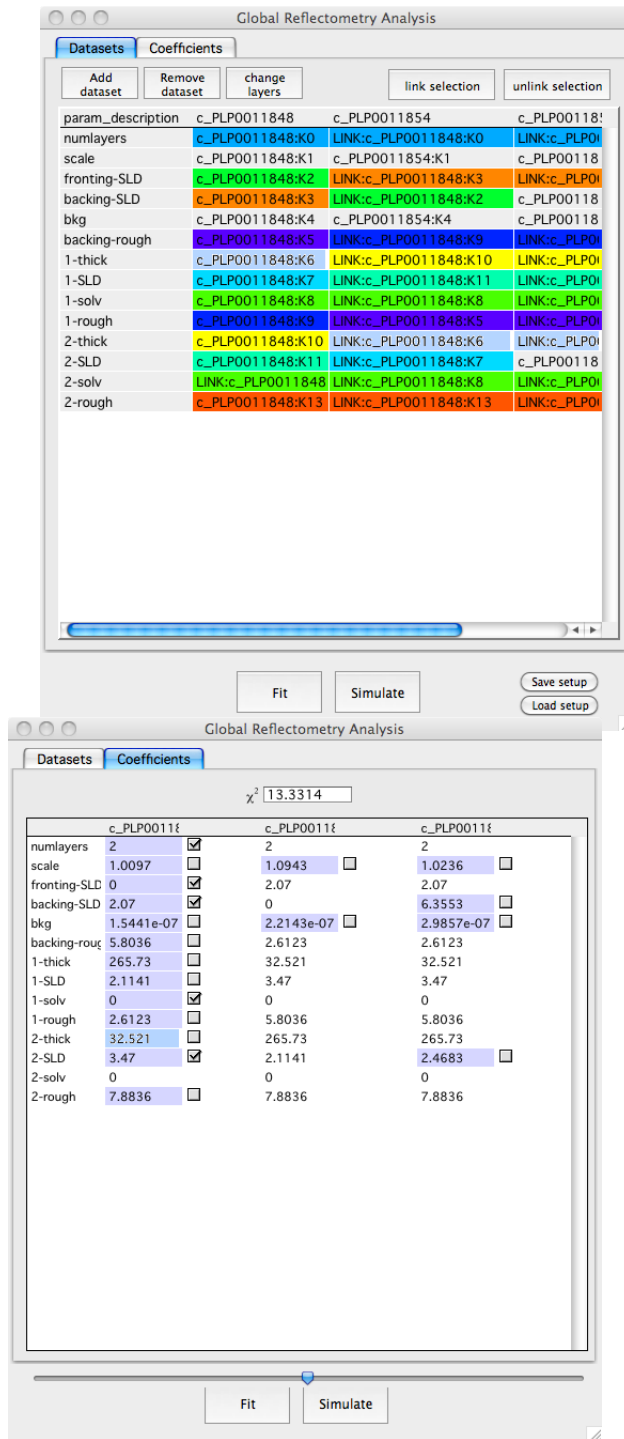
Figure 11: The corefinement panel. In the first tab you setup which datasets you want to co-refine and their linkages. In the second you enter parameter values. Note the slider at the bottom to continuously vary parameters.
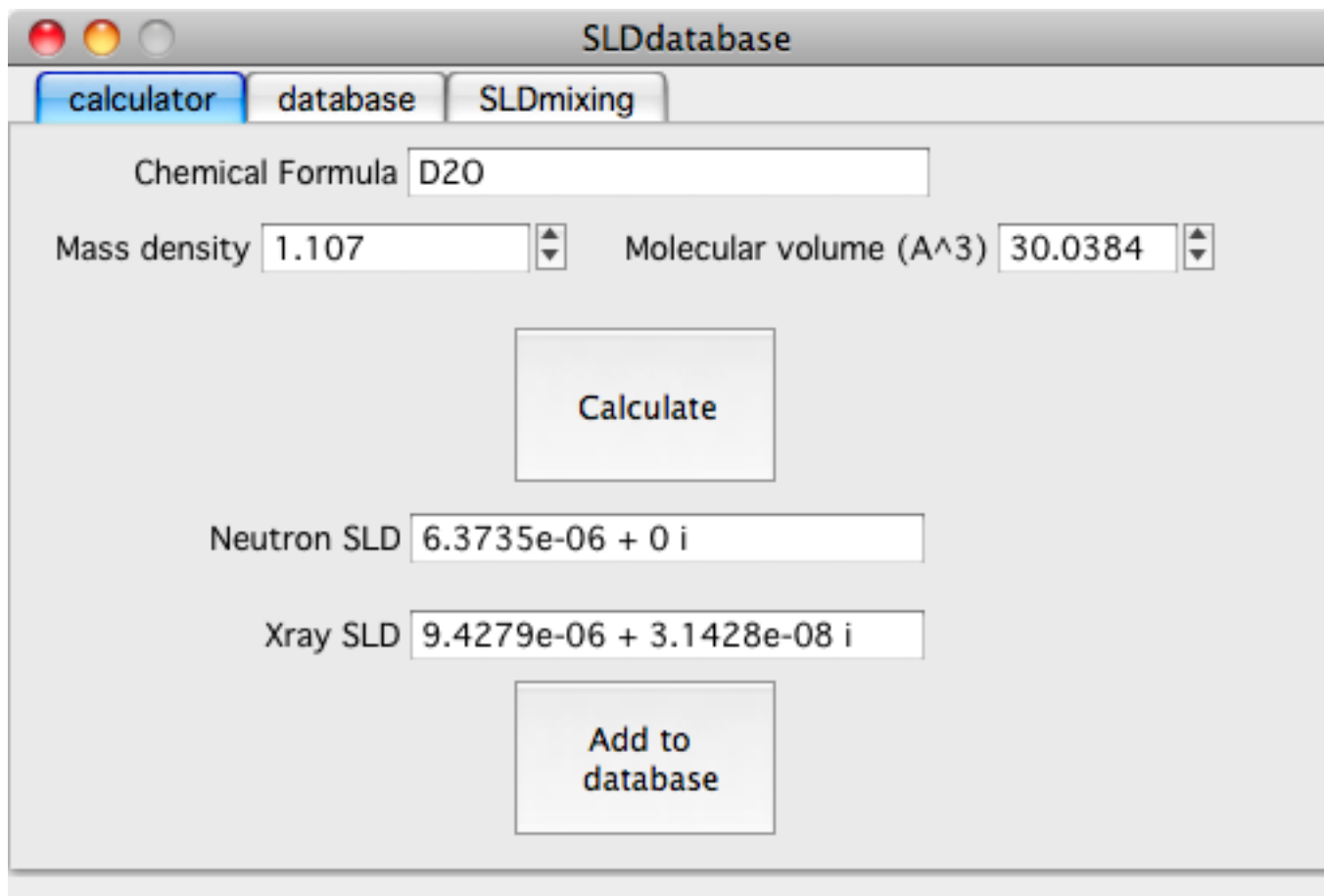
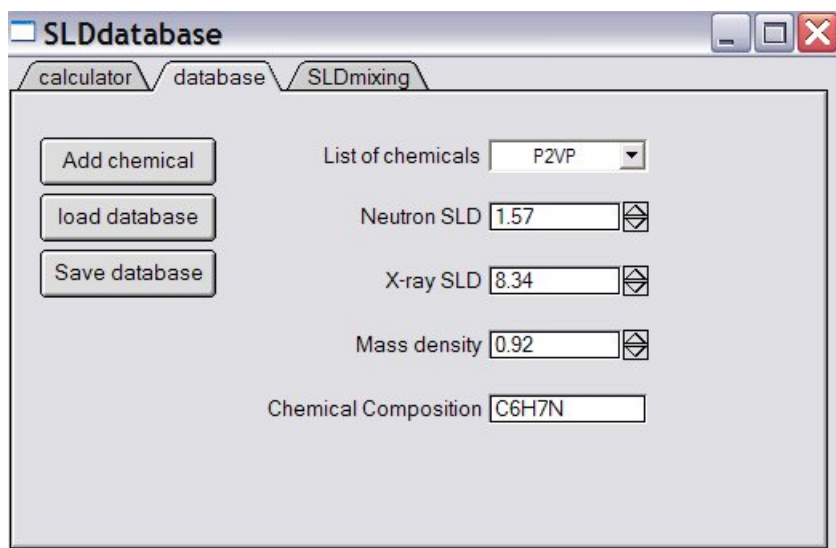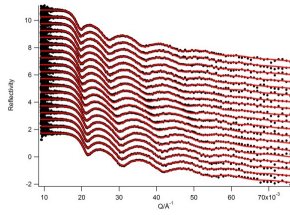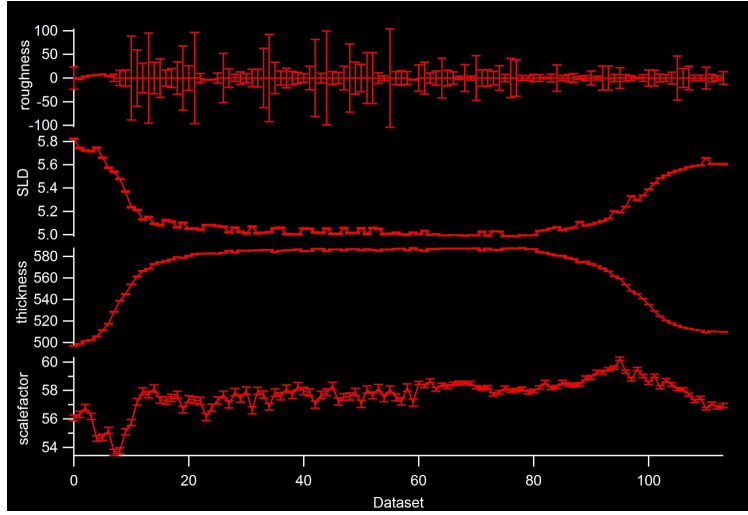Figure 12: The scattering length density database panel

Figure 13: The SLD database holds a list of common chemicals

(a) These datasets were fitted
in <10 seconds

(b) The output from the entire batch fit process

Figure 14: Batchfitting is easy in Motofit. The x-axis corresponds to the dataset number,. The y-axes are shown in reverse order (lowest parameter number is displayed at the bottom.
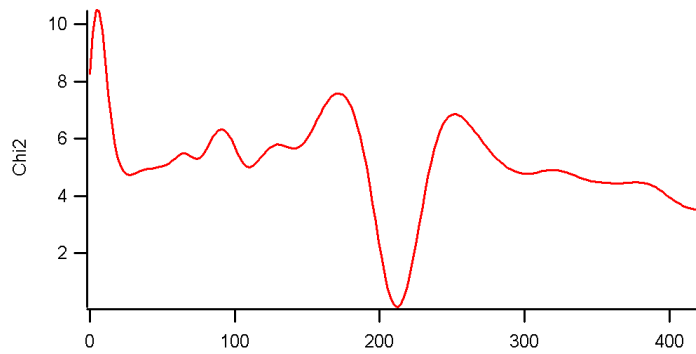

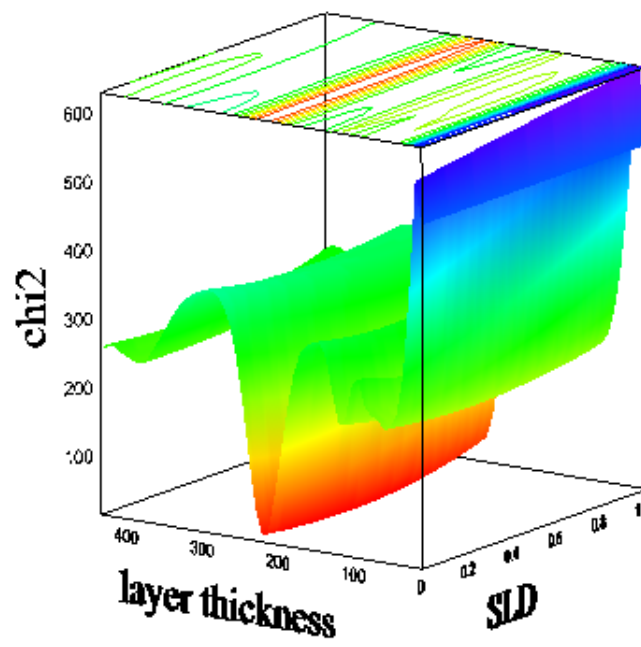Figure 15: Layer thickness Chi2 map for a sample with well defined Kiessig Fringes (212A)

Figure 16: A 2D chi2map of layer thickness and SLD