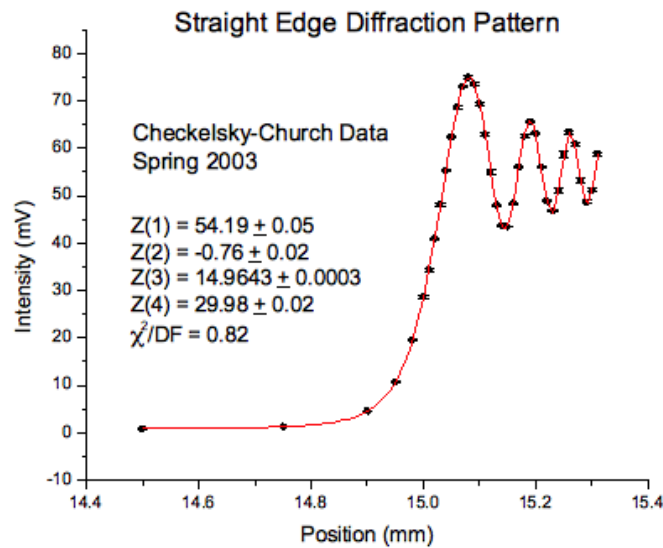

Data Analysis Using MATLAB

MATLAB is a software package for scientific computation and visualization and is produced by The Math Works, Inc. At HMC, MATLAB is used on PCs, Macs, and Unix/Linux workstations. MATLAB's Optimization Toolbox includes a routine LSQCURVEFIT for nonlinear least-squares fitting. While the default fitting algorithm is a “subspace trust-region method based on the interior-reflective Newton method”, the Levenberg-Marquardt algorithm is available as an option. It is reassuring to note that both algorithms give identical results for the Checkelsky-Church data (see below). Fabio Rojas and David Painter (both HMC '96) and Kevin Moore ('99) wrote a MATLAB graphical user interface (GUI) called MatFit that facilitates the fitting of straight edge data to the standard function involving Fresnel sine and cosine integrals. The GUI is really just a collection of “M” files, so if you are familiar with MATLAB you can extend the GUI to fit circular aperture data to an expression (Eqn. (1-1)) containing the first Bessel function.

MATLAB 2011a is available on all of the computers in the lab, in particular the computer associated with the diffraction setup. Before you begin fitting your data in MATLAB, you will want to perform two file manipulations. First, create a text data file (but give it the extension .dat rather than .txt) and copy it to the working subdirectory c:\matfit. Your data file should consist simply of three columns of numbers: position, intensity, and uncertainty in intensity. Example data collected by Candace Church and Joe Checkelsky (both HMC '04) are contained in the file candace_joe.dat. (See the plot below.) Second, you will want to edit the file razorfun.m to set the distance s from the razor to the detector equal to the value appropriate to your setup. Candace and Joe used $s = 1935$ mm. To edit razorfun.m, just locate the file icon in c:\matfit and double-click on it. Wait *patiently* for the MATLAB editor to load and then perform the editing and save



the edited file. More will be said below about the file `razorfun.m`.

To begin your MATLAB fitting session, select Start—All Programs—MATLAB R2011a or double-click on the MATLAB 2011a shortcut on the desktop. Now set the default directory to `c:\matfit` by typing into the MATLAB command window:

```
cd c:\matfit
```

Now run the MatFit “M” file to enter the MATLAB GUI environment written by Rojas/Painter/-Moore. Type into the MATLAB command window:

```
MatFit (Caution: case-sensitive!)
```

A plot window appears with a number of menus laid out across the top of the window.

Data File—Load Datafile Click on the “Data File” menu and highlight the “Load Datafile” menu item. Find your data file in the pop-up directory window, then double-click on it. Your data should now be plotted in the plot window.

Fitting—Fit Data Click on “Load Function File” and double-click on the file `razorfun.m` in the “Files” list. Also click in the “Initial Values for Fitting Parameters:” box, and type, for example, “`Z = [55 0.5 15 30]`”. (That’s a *capital* Z! See below for a discussion of these initial values.) Click on “Start Fit”.

At this point a few words of explanation are in order. The file `razorfun.m` defines the fitting function:

$$f(x) = I_{\text{final}} * 0.5 * ((C+0.5)^2 + (S+0.5)^2) + I_{\text{zero}}$$

where C and S denote the Fresnel cosine and sine integrals with argument

$$\text{arg} = (x - x_{\text{zero}}) * \text{sqrt}(2 / \text{lambda} * (1/s + 1/s_{\text{prime}}))$$

The fitting parameter s_{prime} is the distance between the spatial filter aperture and the razor blade - Candace and Joe measured this to be 30 mm. The fitting parameters are contained in the vector $Z = [I_{\text{final}} I_{\text{zero}} X_{\text{zero}} s_{\text{prime}}]$, so you should enter their initial values in the appropriate order.

After a few seconds the Marquardt algorithm will converge (assuming your initial values make sense), and the fitted curve will be added to the plot. (Move the MATLAB R2011a window around to see the MatFit v1.0 window.) You can monitor the progress of the fitting procedure in the MATLAB R2011a window. The column labeled “f(x)” contains the value of chi-square at each iteration. When convergence is achieved, you are asked in the command window to name your plot of the residuals which will appear in a separate window. Enter, for example, “Normalized Residuals.” Next enter a label for the x-axis of the plot of residuals, e.g., “Position (mm).” Then enter a label for the y-axis, “(Fitted Point - Data Point)/sigma”.

Back in the MatFit v1.0 window, add some information to the plot of the data and the fit.

Plot Options—Show Fitting Results The final values of the fitting parameters, their uncertainties, and chi-square/degree of freedom will be pasted into the plot. Just move the cross-hairs to the point on the plot where you want the upper-left corner of the text box to be, and then left-click with the mouse.

Plot Options—Change Plot Settings You can label the axes and add a title. Click in the “Title” box and type “Straight Edge Diffraction Pattern”. Click in the “x-Axis Label” box and type “Position (mm)”. Click in the “y-Axis Label” box and type “Intensity (mV)”. If you wish, you can add a comment in the “Placeable Text” box, e.g., your name and date. Click on “Done”. Then position the cross-hairs where you want the upper-left corner of your “placeable text”, and left-click with the mouse.

Plot Options—Print Plot You can obtain a hardcopy of your plot on the Scandal_DeskJet6940 printer located in Jacobs B121.

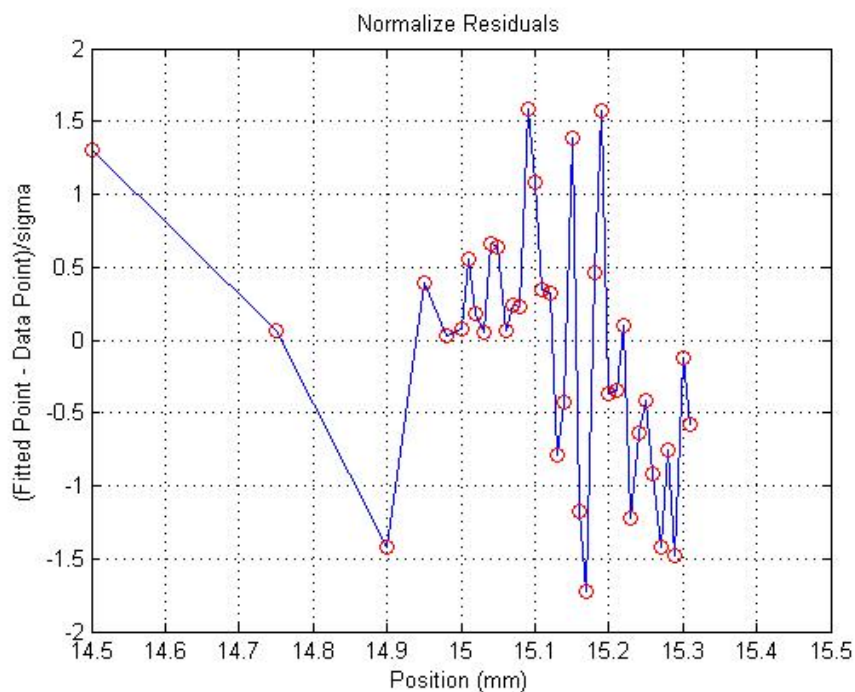
In the Figure No. 2 window,

File—Print Use the standard MATLAB utilities to print a hardcopy of your plot of the normalized residuals on the Scandal_DeskJet6940 printer located in Jacobs B121.

Of course you can always use the standard MATLAB utilities to save the plot of the normalized residuals as a jpeg or bitmap or other graphic format. Unfortunately that is not true of the plot of the data with fitted curve because that plot is not a standard MATLAB figure. An alternative is described below.

During the fitting procedure, MatFit saves the fitted curve to a file named “FittedPoints.txt” in the c:\matfit directory. The file “FittedPoints.txt” may be readily imported into Origin as a single ascii file, and you may then combine the data for the fitted curve with your experimental data

points. When you have combined all of this data into a single Origin worksheet, you may reproduce in Origin the plots you created in MATLAB, both data plus fitted curve and also the plot of normalized residuals. These plots can then be copied and pasted into a Word document. Below we simply used standard MATLAB utilities to save a .jpg version of the Normalized Residuals plot in Figure No. 2, and then inserted the figure here:



To exit the program, return to the MatFit 1.0 window and select Quit—Quit MATLAB.