

Datasqueeze 3.0.4 Manual

www.datasqueezesoftware.com

February 7, 2015



Contents

Contents	2
Introduction.....	4
Downloading and Installing Datasqueeze.....	5
Installation on a Windows PC.....	5
Installation on a PC running Linux	6
Installation on a Macintosh	11
Installation on Solaris or Other Unix-like Operating System	12
Running Datasqueeze.....	13
Getting Started.....	13
Features	14
Sample Session	68
Setting Calibration Parameters.....	68
Look at some real data	74
Help With Least Squares Fits	79
Getting Started with Least-Squares Fits.....	79
Advice on Least Squares Fits.....	83
Technical Details.....	87
Functions Provided.....	89
Scripted (“Batch”) Mode	109
Overview	109
Commands.....	110
Formats used in Batch Commands.....	137

Version History	139
Frequently Asked Questions	141
Installation	141
File Types	142
Calibration/Centering	143
Plots	144
Least-Squares Fits	145
Run-time errors	146
Feedback	148
Note on Copyrights	148
About the Author	148

Introduction

Welcome to Datasqueeze, a graphical interface for analyzing data from 2D x-ray detectors. Version 3.0.4, released February 2015, is designed for analysis of data collected by many different types of detector. It implements facilities for changing the color scale of a false color image, drawing constant-intensity contours, recentering the image, correcting for a tilt of the detector with respect to the beam normal, changing the q-scale of the entire image, producing x-y plots versus q, 2-theta, qx, qy, or chi, saving the image in multiple graphics formats, saving the x-y data as an ASCII file, and adding or subtracting multiple data files. The data extracted for the x-y plot can be least-squares fit to a variety of commonly used functions. A summary page can be printed containing most useful information about the dataset including the false color image and a line. The recentering and rescaling can be done using a point-and-click graphical interface, by putting in previously calculated numbers, via least-squares fit to a Bragg ring of known d-spacing, or by using a Bruker calibration file (*.std). The program normally operates using a graphical user interface (GUI), but can also be operated in a “batch” mode for automated processing of multiple files.

This program may not be copied or redistributed in any form without the express permission of Datasqueeze Software.

This manual collects information on using Datasqueeze that is available via the onboard Help menu and the web page (<http://www.datasqueezesoftware.com>). It also contains screenshots and other images not available in the onboard manual. It may be useful for those who find paging through a printed or PDF document more useful than scrolling through an html-based document.

Downloading and Installing Datasqueeze

The trial version of Datasqueeze is downloaded at <http://www.datasqueezesoftware.com/download.html>. It can be run free of charge for up to ten days. To run the program after the trial period, point your browser to www.datasqueezesoftware.com/purchase.html to purchase an access key.

Installation on a Windows PC

Requirements:

- A PC running at at least 300 MHz.
- A color monitor display resolution of at least 1024 x 768.
- At least 60 MB of available memory.
- At least 10 MB of free disk space.
- The Windows Operating System (XP, NT, Vista, 7, or 8).

Installing Datasqueeze

- You need to make sure that you are logged in as administrator, or that you have administrator privileges.
- If you are not sure whether you have Java 1.7 or above installed (*JavaSoft* does **not** count) point your browser to <http://java.com/en/download/index.jsp> to download the latest version of Oracle Java. (Note: if, at a later stage in the installation process, you get a message that says something like “Could not find main class” it means that your current version of Java is not sufficiently up to date).
- Point your browser to <http://www.datasqueezesoftware.com/DSInstall.exe> (6 MB)
- If your browser did not automatically initiate the installation process, open the installation package, which is quite likely located on your Desktop and is called "DSInstall.exe", and double-click on the DSinstall icon. You will be guided through the installation procedure
- You can start Datasqueeze either through the Program Files menu, or by the green and orange icon that the installer should have left on your desktop.
- When you first run, you will be asked to enter an access code and to agree to some legal stuff. If you do not have an access code, you can run the Trial version, for up to ten days. If, after downloading and running the trial version, you decide you want to upgrade to the full-featured registered version, please go to www.datasqueezesoftware.html/purchase.html to purchase the access key-this is all you will need to make the upgrade.
- An onboard help menu provides additional instructions; new users are strongly urged to read the help file (or the corresponding sections in this manual) and the

tutorial before using the program. (Most components in the graphical interface can also be right-clicked for a short description).

- When you have completely installed the program, you can drag the installer to the Recycle Bin. If things do not initially work as you expect, please contact support at support@datasqueezesoftware.com.

Contents

When you look inside the Program Files→dataqueeze folder you should see the following items:

- A file with the green and red Datasqueeze icon called Datasqueeze.exe. This can be double-clicked to launch the application (which should also be accessible from your Start Menu).
- A file called `datasqueeze_manual.pdf`. This is the manual you are reading--you could copy it anywhere convenient or print it out if you wanted.
- A folder called Samples. Within it are:
 - Files called `agbe_calib.unw` and `sample.unw`. These are "typical" Bruker-Siemens data files for you to play with.
 - A file called `agbe.std`. This is a typical Bruker calibration file, for silver behenate, which you could also use as a template.

Installation on a PC running Linux

Requirements

- A PC running at at least 300 MHz.
- A color monitor display resolution of at least 1024 x 768.
- At least 60 MB of available memory. Note that in some multiple-user environments the system administrator may have placed quotas on the amount of disk space and/or memory per user, so that even if the machine has enough memory you as a user may not; check with your system administrator if you are not sure.
- At least 10 MB of free disk space.
- The Linux operating system. This application has only been tested on a limited number of platforms, but there is no reason at this point to believe that it will not run on all flavors of Linux, although support for printing varies from version to version.

Installing the Java Platform

- This program was written Java 2 Standard Edition, version 1.7. It should be compatible with any later version of Java. If you are an earlier version of Java, or if your computer does not have Java 2, you need to download it, which can be done for free from the Oracle site. The download URL is currently

<http://java.com/en/download/index.jsp> The open-source Java provided with many flavors of Linux will probably **not** work. (Note: if, at a later stage in the installation process, you get a message that says something like “Could not find main class” it means that your current version of Java is not sufficiently up to date).

- You will have a choice of products to download; you want the latest version of the Java Development Kit (JDK).
- You will probably have to be the administrator (superuser) before you do this part.
- If you are not sure whether java has been loaded on your machine or whether it is the right version, type

java -version

at the command line. It should say something like

Java HotSpot(TM) Client VM (build 1.3.1_02-b02, mixed mode

which in this case would indicate that you are running version 1.3.1

- In some cases there may be multiple versions of java installed on the same computer, in which case it might be necessary to type a different command to get the latest version. For example, on a computer which had both 1.2 and 1.4 loaded, typing

java -version

would return

Solaris VM (build Solaris_JDK_1.2.2_07a, native threads, sunwjit

but typing

/pkg/j/j2sdk1.4.0/bin/java -version

might return

Java(TM) 2 Runtime Environment, Standard Edition (build 1.4.0-b92)

Locating Java On Your System

If just typing “*java -version*” does not work, your path settings do not include the location of java and/or it is located in a funny place. Look around for it. You might try typing

```
find /usr -name java -a -print
```

If you think you have found the application, test it out. For example, suppose that you located a copy of *java* in *usr/java/bin/java* (This is a common location). Then in that case typing

```
/usr/java/bin/java -version
```

should give you version information.

Installing Datasqueeze

- Point your browser to http://www.datasqueezesoftware.com/ds_installation.tar.gz to download the installation package
- Locate where your browser downloaded the file called *ds_installation.tar.gz*, and change to that directory. (Some browsers may partially or completely uncompress the file, producing something called *ds_installation.tar* or a folder called *ds_installation*. If so, ignore the next couple of steps).
- Type

```
gunzip ds_installation.tar.gz
```

This should produce a new file called *ds_installation.tar*

- Type

```
tar xf ds_installation.tar
```

- This should produce a new directory called *ds_installation*. Change to that directory (*cd ds_installation*)
- You have two options for installing the program. If you are a system administrator (superuser) you can install it in a central location so that everyone on your system can use it. If you are an ordinary user, you can install it in your home directory.
- For system-wide installation:
 - Type

```
su (CR)
```

to become superuser (you will need the superuser password). If you do not have permission to do this, talk to your system administrator before proceeding.

- This installation assumes that the *java* application lives in */usr/java/bin/java*. To see if this is the case, type

```
ls -ls /usr/java/bin/
```

- If you get something that looks reasonable, you are ready to go. Otherwise, you will need to locate Java on your system as discussed above. Then edit the 5'th line of the file called *ds_install_global* to reflect the true location. (See comments above multiple versions of Java.)

- Type:

```
sh ds_install_global
```

- To start the application, type

```
datasqueeze
```

from whatever directory you might be working in

- For installation as a user application (in your directory only):

- This installation assumes that the java application lives in */usr/java/bin/java*. To see if this is the case, type

```
ls -ls /usr/java/bin/java
```

If you get something that looks reasonable, you are ready to go. Otherwise, you will need to locate Java on your system as discussed above.

- Then edit the 5'th line of the file called *ds_install_local* to reflect the true location. (See comments above multiple versions of Java.)

- Type:

```
sh ds_install_local
```

- To start the application, type

```
/home/yourname/ds/dataqueeze
```

or

```
/home/yourname/ds/datasqueeze file1 file2 file3 ...
```

(where *yourname* is your login name) from whatever directory you might be working in. Note that a new window is opened for each file, consuming memory, so you will not be able to open more than a few files at the same time.

- If Datasqueeze refuses to run at all, it is likely that your current version of Java (or at least your current default version) is too old. Go back to the instructions on installing Java on your system.

- When you first run, you will be asked to enter an access code and to agree to some legal stuff. If you do not have an access code, you can run the Trial version, for up to ten days. If, after downloading and running the trial version, you decide you want to upgrade to the full-featured registered version, please go to www.datasqueezesoftware.html/purchase.html to purchase the access key-this is all you will need to make the upgrade.
- If you have done the system-wide installation, you will need to give the access code to each potential user of the application. Note that your **license does not permit you to make Datasqueeze available to a large user base from a central server**. If it is to be loaded on a central server and accessed remotely it should be only for the use of a small research group (5 users or less).
- **Contents:** if you look inside the `ds_installation`→`dataqueeze_files` folder you should see the following items:
 - A file called `datasqueeze_manual.pdf`. This is the manual you are reading-you could copy it anywhere convenient or print it out if you wanted.
 - A file called `datasqueeze.jar`. This is contains the actual compiled Java code.
 - A folder called `samples`. Within it are:
 - Files called `agbe_calib.unw` and `sample.unw`. These are "typical" Bruker-Siemens data files for you to play with.
 - A file called `agbe.std`. This is a typical Bruker calibration file, for silver behenate, which you could also use as a template.
- An onboard help menu provides additional instructions; new users are strongly urged to read the help file (or the corresponding section in this manual) and the tutorial before using the program. (Most components in the graphical interface can also be right-clicked for a short description).
- When you have completely installed the program, you can remove the installation files:

```
rm -f -r ds_installation.tar ds_installation
```

- If things do not initially work as you expect, please contact support at support@datasqueezesoftware.com.

Note on Running in X-windows Environment

Datasqueeze can be run from a remote computer in an X-windows environment. Since Datasqueeze brings up a new graphics window, appropriate X-windows protocols must be followed. Space does not permit a full description of the X-windows environment, and the exact commands may depend on your particular implementation. But the following instructions should provide a preliminary guide.

Suppose you are running on a Linux machine called *tweedle.xyz.edu* and you have Datasqueeze installed on a X-windows compatible Linux machine called *dum.xyz.edu*. Your login name on *dum.xyz.edu* is *lcarroll*. Starting in a window on *tweedle.xyz.edu* you would type something like:

```
xhost +s dum.xyz.edu.
```

```
slogin -l lcarroll dum.xyz.edu
```

```
(...enter password)
```

```
setenv DISPLAY tweedle.xyz.edu:0.0
```

```
datasqueeze
```

The first command tells *tweedle* that it is OK to accept new windows from *dum*. The second gets you logged in to *dum*. The third command tells *dum* that graphical output should appear on the console (0.0) of *tweedle*. And, finally, the application is started.

Installation on a Macintosh

Requirements

- A Macintosh running at at least 300 MHz.
- A color monitor display resolution of at least 1024 x 768.
- Mac OS 10.6 or above
- At least 60 MB of available memory.
- At least 200 MB of free disk space.

Installing Datasqueeze

- You may need to be the “administrator” for some parts of this. If you do not have permission to do this, talk to your system administrator.
- If you are using version 10.9 (“Mavericks”) or above, point your browser to http://www.datasqueezesoftware.com/ds_installation_jre.dmg.bin to download the installation package. If you are using version 10.8 or below, point your browser to http://www.datasqueezesoftware.com/ds_installation_jre.dmg.bin.
- Double-click the .dmg file that appears to open the disk image.
- From inside the *ds_installation* folder, the *datasqueeze.app* file (should look like an icon of two hands squeezing an orange) to the Applications folder on your main hard drive (probably called Macintosh HD).
- When you first run, you will be asked to enter an access code and to agree to some legal stuff. If you do not have an access code, you can run the Trial version, for up to ten days. If, after downloading and running the trial version, you decide

you want to upgrade to the full-featured registered version, please go to www.datasqueezesoftware.html/purchase.html to purchase the access key-this is all you will need to make the upgrade.

- An onboard help menu provides additional instructions; new users are strongly urged to read the help file (or the corresponding section in this manual) and the tutorial before using the program. (Most components in the graphical interface can also be control-clicked for a short description).
- When the installation is completed you can drag *ds_installation.dmg* the disk image, and the *ds_installation* folder (if they all still exist) to the Trash.
- If things do not initially work as you expect, please contact support at support@datasqueezesoftware.com.

Contents

If you look inside the Applications→dataqueeze_files folder you should see the following items:

- A file called *datasqueeze_manual.pdf*. This is the manual you are reading--you could copy it anywhere convenient or print it out if you wanted.
- A folder called Samples. Within it are:
 - Files called *agbe_calib.unw* and *sample.unw*. These are "typical" Bruker-Siemens data files for you to play with.
 - A file called *agbe.std*. This is a typical Bruker calibration file, for silver behenate, which you could also use as a template.

Installation on Solaris or Other Unix-like Operating System

Follow the instructions for Installation on a PC running Linux. You are somewhat on your own here. The application has undergone minimal testing on a Solaris system (where it was at least observed to run, read in data, etc.) but has not been extensively debugged. It has not been tested on other Unix systems, but there is a good chance that it will work the same way.

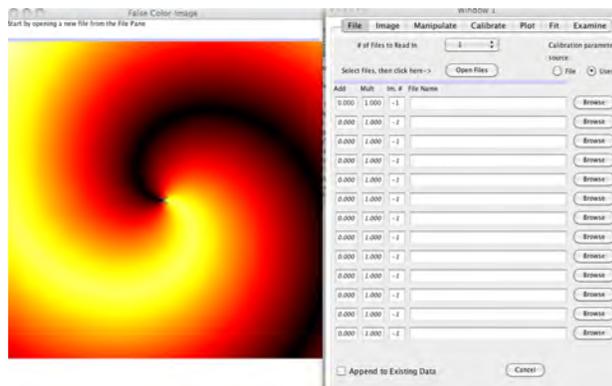
Running Datasqueeze

Getting Started

The program is started in slightly different ways depending on the platform you are running. On Macintosh or Windows, double-click on the “*Datasqueeze*” icon (two hands squeezing an orange) if it is visible on your desktop. If the icon is not visible, you should be able to find it in the Start/Programs menu (Windows) or the Applications folder (Macintosh). On Linux, type *datasqueeze* at a shell prompt.

If you have not previously run the program, you will see a window that requests an access key. If you do not have a valid key, click on “Choose Trial Mode” to run the Trial version. You will then be asked to agree to various legal statements, the most important of which is that you will not redistribute the program without permission. You will then be asked a variety of questions about your system: your preferred number format, the typical configuration of your instrument, the wavelength expected for your incident beam, the detector pixel size in microns, the detector width or diagonal span in centimeters, and the sample-detector distance. If you don’t know some of these parameters don’t worry; except for the wavelength they are not necessary for the calibration.

You should now see an abstract spiral image on the left side of the screen, a window with some buttons and slide bars on the right side of the screen, and a white panel with some writing at the bottom. The red-yellow spiral is where the false color image will appear. The button window is a tabbed panel that allows you to manipulate the data and make plots. The white panel at the bottom is where x - y plots will appear.



Note that many program components have an individual help message associated with them. If you don’t know what a component does, right click on the component (control-click on Macintosh). In many cases, selecting the box with the word Info will bring up more information on that component.

Before doing anything else, you have to read in data. You read in a file or multiple files using the File panel as described in more detail below. You then process the information using the other tabbed panes.

Features

Command Log

Immediately above the tabs labeled Image, Plot, etc., there is a small window which often reports the current status of the program, such as the name of the last data file opened. Look here for error messages if things seem not to be working as you expect. If the window turns red then there has been some kind of error.

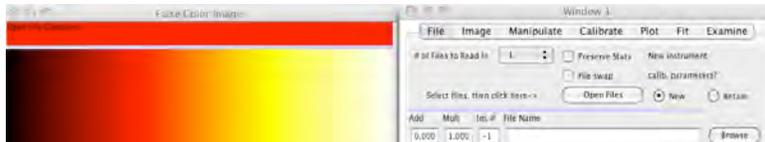
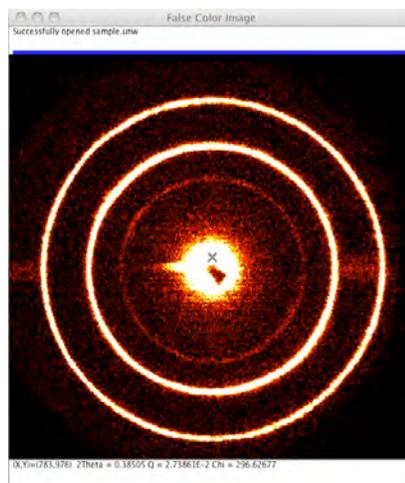


Image Window

This is where the false-color image of the data appears. The color scale ranges from black (least intense) to white (most intense). An X marks the believed position of the center of the image. (In the less-commonly-used Wide Angle Mode this X is replaced by an arrow pointing in the direction of increasing 2θ). Almost all features of the image are actually controlled from other panels, although occasionally circles and drag buttons will appear on the false color image window as described in the other sections. The image created can be saved in a graphics file.



Note that some commands cause the false color image window to be recalculated and redrawn, and that this can be a fairly time-consuming process on slower machines. The program has to look at between one and seventy megabytes (depending on the file format) of data, do some kind of calculation on each data point, and convert this information to a 512 x 512 array of colors.

A small window just below the image window tracks the current cursor position in several sets of units: the (x,y) coordinates of the data array, the q -value ($= (4\pi / \lambda) \sin \theta$), 2θ , and χ . Nothing is shown if no data have been read in. The angular and q values are meaningless if the image has not been calibrated. Note also that, depending on the operating system and hardware used, the tip of the cursor may not lie exactly over the pixel reported.

File Panel

This panel allows you to open one or more data files. To rapidly open just one file with the default options, use the Open item in the File menu. For a quick open using the File Panel, leave the number of files at the default of 1. Click on the Browse button to select the file to be opened. Once the file has been selected, click the Open Files button to actually open the file. After a pause, a false color image should appear to the left. A detailed description of each component of this window follows.

- **# of Files to Read In:** This drop-down menu allows you choose the number of files to be summed, from 1 to 13 (default 1). If the number is changed, the number of enabled “open file” dialog boxes changes. If you want to sum more than 13 files you can use the “Append” box (see below).



- **Calibration Parameter Source:** To properly interpret a diffraction image, you need to know the instrument parameters, for example wavelength and sample to detector distance. In some cases these are contained as metadata within the data file, but in most cases they must be determined by the user (using methods in the Calibrate window). The “File” and “User” radio buttons allow you to decide whether you want to use user-determined values (the default) or metadata from the file. The User button will be enabled automatically during the calibration process.)



- **Open File Dialog Boxes:** There is one of these for each file to be opened. The text box should contain the full path name of the data file to be opened. You can type this in, paste it in, or most commonly use the Browse button to the right.
 - “Add” is the number of counts to be added to each pixel (default zero). Most commonly for background subtraction it is likely to be a small negative integer.
 - “Mult” is the value by which each pixel will be multiplied after being read in. “Mult” can be negative, for example if this is a background file.
 - “Im #” is the index of the image within the file. Some data file formats allow multiple images within one file. If “Im #” = -1 (the default) then all images within the file are summed, and “bad” data points (“zingers”) are thrown out. If “Im #” is greater than or equal to zero, then just one image is opened. It is assumed that the numbering of the images always starts



with zero, no matter what the convention might be for a particular file format.

- **Browse button:** Use the Browse button to select the individual file. In addition to a listing of files in the current folder, you will get a drop-down menu with different detector types. Select the one appropriate for the kind of file you are looking at. If you have no idea what type of detector your data came from, select “Unknown.” The program will try each in turn until it finds one that works (this is somewhat slow).



- **Open Files:** Click this after all the open file dialog boxes are set correctly. Nothing will be read in until you do this. If everything goes well, you will see a message in the command log, and a new false color image will appear. If there was a problem, the offending dialog box will be highlighted in red and an error message will appear in the command log.



- **Order of Operations:** First, the contents of the Add box are added to each pixel in a given file. Then, each pixel is multiplied by the Mult box contents. Finally, the pixels in all the files are added together. Thus, if you had a data file *A.raw* which was counted for half as long as a background file *B.raw*, you might set the Add values for both files to be zero, the Mult value for *A.raw* to be 2, and the Mult value for *B.raw* to -1.

- **Append:** Normally, each time new data are read in, the data already in memory are overwritten. Checking this box adds the new data on top of the existing data. This allows, in principle, an arbitrary number of files to be added together. Do not forget to uncheck when you want to start analyzing a completely new data set. In Batch mode see the APPENDTOFILE option.



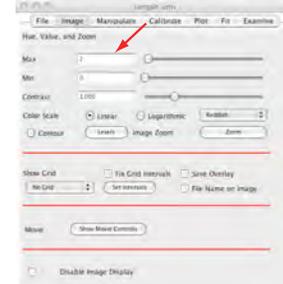
- **Cancel Button:** If you realize that you made a mistake in selecting the files you want to open, cancel the opening process by clicking this button.

In Batch mode see the READFILE option.

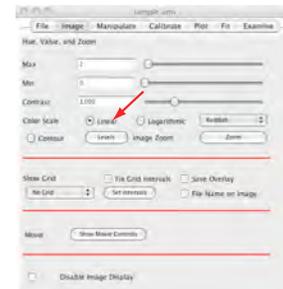
Image Panel

This panel is selected by clicking on the Image tab in the left-hand window. It contains sliders, buttons, and windows for changing features of the false-color image. Most of these will only work after a data file has been opened.

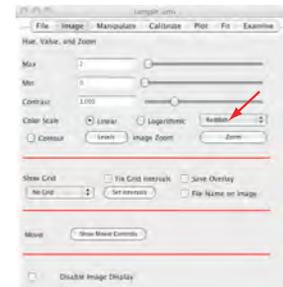
- **Value and Contrast:** “Black” corresponds to pixels with a small number of counts and white to pixels with a large number of counts, with a range of hues in between. You can change the maximum and minimum values (white and black) in two ways: by dragging the two sliders to the right or left, or by entering in the desired values in the boxes to the left of the sliders. The boxes are probably preferred to the slider unless you have a very fast computer, since the program recalculates the image every time one of these values changed, and things can be jerky with the slider. You can also change the contrast using the contrast box or slider. In Batch mode see the MINVALUE, MAXVALUE, and CONTRAST options.



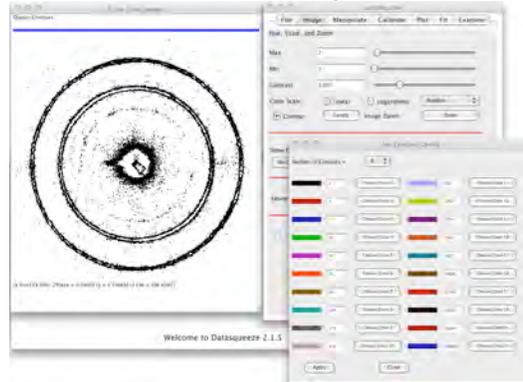
- **Linear/Logarithmic:** Buttons just below the slider allows you to change the nature of the color scale. If set to Linear (default), the color scale is linear with the counts/pixel from the smallest to the largest. If changed to Logarithmic, the color is proportional to the logarithmic of the intensity. In Batch mode see the LINEARSCALE and LOGSCALE options.



- **Color Scheme:** The “Color Scheme” drop-down menu allows you to specify the rainbow of colors on going from black to white--more red, more green, etc. Each number corresponds to a different color. The last two options are black-and-white only, which may be useful if the image will ultimately output to a black/white printer. “White/Black” produces an image similar to the previous ones, with whiter pixels corresponding to more intensity. “Black/White” produces an inverted image, with blacker pixels corresponding to more intensity--this one looks very much like old-fashioned photographic film. In Batch mode see the SETCOLOR option.



- **Contour:** This is an alternate way of displaying the data, replacing the false color display controlled by the color scale and color scheme. Constant intensity contours are drawn at selected intensity levels. The resulting image can be zoomed or the normal way.

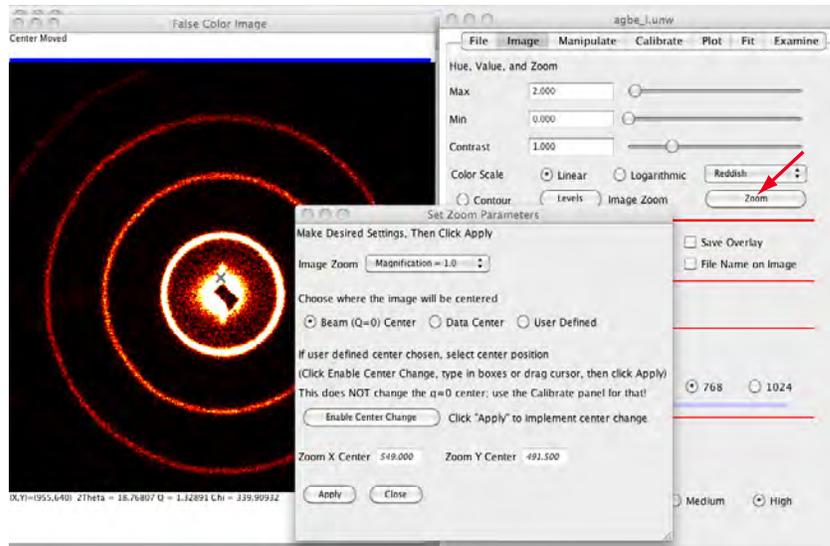


Clicking on the Contour button switches from Linear or Logarithmic mode to contour mode. The default is a set of logarithmically spaced contour lines. The number of contours, intensity value for each contour, and color of each contour line can be altered in a new window that is opened using the **Levels** button. The contours will be immediately redrawn if you click the **Apply** button, assuming that you have already clicked the Contour button--otherwise, the changes will take effect the next time you go into contour mode. If you decide to play with the contour levels and/or colors, keep the following in mind:

- The program assumes that the intensity levels are in increasing order. Arranging them any other way may lead to unpredictable results.
- The time to draw the image is approximately proportional to the number of contour lines chosen. You will see a real decrease in performance if you select too many contour lines.
- Too many contour lines can also make an image too busy.
- Adjacent contour lines should have very different colors, otherwise you will not be able to determine where the changes happen.
- You cannot change the parameters of a contour line whose index is greater than the maximum number of lines to be drawn.
- The ultimate information of the data is limited by the pixelation. Things may look a little odd at extreme magnification. Low-statistics data will result in very noisy, fractal-looking contour lines, because the contour lines faithfully follow the contours of the data including statistical fluctuations.
- If you are looking at broad features (which is typically the case if you are using contours at all) consider using the Condense option to bin together pixels, resulting in better statistics but poorer resolution.

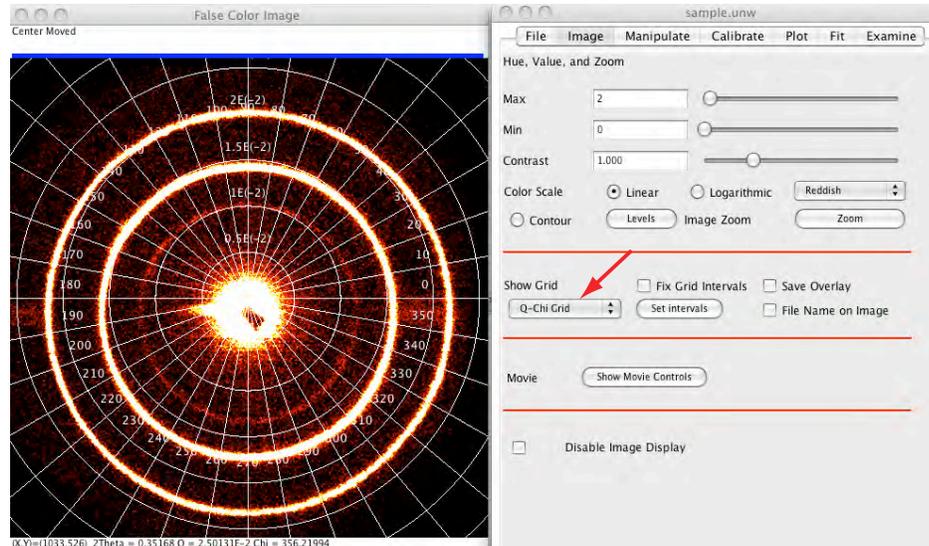
In Batch mode, see the CONTOURSCALE, NCONTOUR, and SETCONTOUR options.

- **Zoom:** Selecting this button brings up a new window that provides further control of the magnification of the image.



- **Image Zoom:** This drop-down menu changes the magnification of the false color image, in factors of 2, from 0.5 to 32. If Zoom=1 is set, the entire data set appears in the image; if Zoom=2 then only the center data pixels are shown, etc. If Zoom=0.5 the image is demagnified 50 percent.
- **Image Centering:** There are three options provided.
 1. **Beam (Q=0) Center:** If this option is selected, the image will always be magnified in such a way that the center of the false color image is the place where the direct beam hits the detector (usually covered by a beam stop). This is probably the most useful choice under normal operating conditions. However, it fails when the beam zero does not lie on the detector, which may happen when operating in wide-angle mode. Setting this center is most often accomplished using the tools in the Calibrate panel, but can also be done manually in this window. Click “Enable Center Changed”, drag the blue cursor or type in the boxes labeled “Zoom X Center” and “Zoom Y Center” until the desired position is selected. Then click “Apply”.
 2. **Data Center:** If this option is selected, the image is magnified in such a way that the image of the false color image is the geometrical center of the data set. This is most useful when the beam zero is near the edge of the detector or completely off the detector.
 3. **User Defined:** If this option is defined, the user can choose an arbitrary spot on the image to use as the magnification center. Click “Enable Center Changed”, drag the blue cursor or type in the boxes labeled “Zoom X Center” and “Zoom Y Center” until the desired position is selected. Then click “Apply”.
- **Apply:** None of these changes will take effect until you click the Apply Button.

- **Close:** Click the Close button to close the Zoom window without making further changes.
- **Show Grid:** This drop-down menu allows you to superimpose a grid on the false color image, for better locating the positions of features of interest. Possibilities are: No Grid (default); Q-Chi Grid (circles are in equal increments of momentum transfer, radial lines every ten degrees of azimuth), 2-Theta-Chi Grid (circles are in equal increments of scattering angle), or Qx-Qy Grid (square or rectangular grid). Notice that the grid is meaningless unless the calibration constants are set correctly (range, center, tilt, etc.). In Batch mode see the SETGRID option.

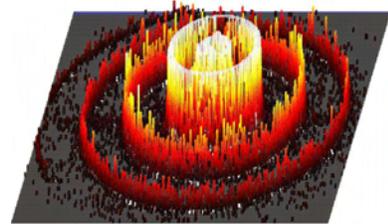


- **Fix Grid Intervals:** This checkbox affects the operation of the Show Grid command. If it is unchecked, then the program will calculate for you the positions of the grid circles and lines. If it is checked, then a new window appears which allows you to set these values yourself. In Batch mode see the FORCEGRID and GRIDINCREMENTS options.
- **Set Intervals:** This button brings back the window that was opened by the Fix Grid Intervals box, so that you can once again set the values of the grid overlay. Note that these values will have no effect unless the Fix Grid Intervals box is checked.
- **Save Overlay:** If this box is checked, then any "overlays" on the false color image (grid, the X at the center, region selected for plotting) will be copied onto an image copied into the clipboard or saved as a graphics image. In Batch mode see the SAVEOVERLAY option.
- **File Name on Image:** If this box is checked, then the name of the currently open file will be displaced on the false color image. This might be useful if you were pasting the images into another document and wanted to keep track of which was which. In Batch mode see the WRITEFILENAME option.
- **Show Movie Controls:** This option allows you to make a movie (or a fast slideshow) of false color



images. It is suggested that you begin by opening an image, setting the color scale the way you want it, and clicking on “Disable Automatic Image Rescale.” This will ensure that all frames show with the same maximum/minimum. Click on “Show Movie Controls” to bring up a new window. “New Movie” initializes the sequence. Each time you have a false-color image that has been read in and optimized the way you like it, click on “Add Frame” to add that image to the buffer. When you have all images you want, click “Play” to show the sequence and “Stop” to stop showing the sequence. Click on “Save” to save the final result as a .avi file. avi files are readable by most movie players, and can also be imported into an application such as Powerpoint for presentation purposes. The drop-down menu shows the nominal delay between frames, in milliseconds. The actual time may vary depending on the speed of your computer and how many other processes are running. The movie option can also be accessed programmatically in the Batch mode using the INITIALIZEMOVIE, ADDMOVIEFRAME, and PLAYMOVIE options. You can also add frames as one of the options under Process Multiple Files, and you can save a movie using the File->Save Movie menu item.

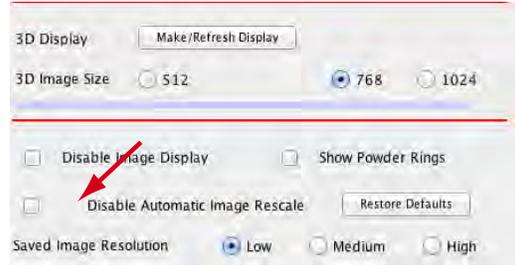
- **3D Display** Instead of the standard false color image or contour image, it is possible to make a pseudo-three-dimensional image of the raw data. After some binning, a series of line plots are superimposed to make “mountains” and “valleys” in a separate window. You must first read in data and if desired set the hue, value, and zoom in the top part of the Image panel. Then select the image size (in pixels) using the appropriate button next to “Image Size”. Finally, click “Make/Refresh Display”. You will need to click again each time you wish to refresh this image. Note that the peaks are truncated at the value set by “Max” so the image will have a different appearance depending on the Max/Min/Color Scale settings. The generated image can be saved using File -> Save 3D Image. In batch mode see MAKE3DIMAGE and SAVE3DIMAGE.

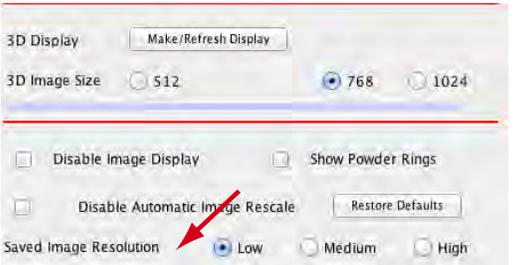


- Disable Image Display** Normally, the false color image is displayed and updated every time an important change is made (new data, change in color scale, change in center, etc.) The false color image can be disabled by checking the “Disable Display” box. This is normally used only if data are being analyzed in batch mode, in which case the operator does not plan to actually look at the images but speed is important. Accessed programmatically using `IMAGEDISPLAYENABLE`.


- Show Powder Rings:** If this box is checked, then when powder diffraction lines are displayed on the plot image (using the "Show Powder Diffraction Line Positions" option in the Plot panel) the corresponding Bragg diffraction rings will also show up on the false color image.


- Disable Automatic Image Rescale** Normally, Datasqueeze resets the color scale (maximum-minimum) of the false color image each time a new file is opened, attempting to choose a sensible scale that will highlight the features of interest. If the “Disable Automatic Image Rescale” box is checked, this feature is disabled, and the same color scale is used. This might be useful, for example, when comparing a series of similar files or when making a “movie” in this case the user might wish to use the same scale throughout the series. No matter how the check box is set, the *first* image opened in a session is always autoscaled. This feature is accessed programmatically using `IMAGEAUTORECALCENABLE`.


- Restore Defaults** Restore false color image settings to default values. This feature is accessed programmatically using `RESTOREDEFAULTIMAGEVALUES`.
- Saved Image Resolution** This radio button sets the resolution of a false-color image saved with the File->Save False Color Image or Process Multiple Files options. “Low” resolution corresponds to a 512x512 image, identical to that displayed on the screen. “Medium” resolution corresponds to a 1024x1024 image. A “High” resolution is saved with the same number of pixels per side as the larger dimension of pixels in the original data set—i.e., a 1024x2048 data image would be saved as a 2048x2048 square pixel array.



Note that higher resolution images take longer to save and may consume substantially more disk space.

Manipulate Panel

This panel is selected by clicking on the Manipulate tab in the left-hand window. It contains buttons that allow you to modify or smooth the data in various ways. Note that most of these options actually change the data in memory, and there is no “undo” feature—if you are not happy with the results you will need to read the data in again. (However, your original data file is never modified). Also, it is important to calibrate the pattern (by setting the range, center, tilt, etc.) *before* modifying the data by condensing, etc. This is because most of the “Manipulate” features also result in changes to the calibration parameters that are not retained when the next file is opened, or saved for future use.

- **De-Zing:** This button, if clicked, will remove “zingers” from the image. These are pixels that have substantially (and astatistically) more intensity than all of their neighbors. (Specifically, pixels that deviate by more than five standard deviations from all four of their nearest neighbors). This can be a problem particularly with some CCD detectors. The pixel is replaced with the average of the surrounding nearest-neighbor pixels. Note that this process is irreversible, although the original data file is untouched. If you are unhappy with the results, you will have to read in the data again. Note also that the algorithm may falsely identify pixels as “zingers” if you have very sharp Bragg peaks, with widths on the order of several pixels. The dezing-ing process can take several seconds; a progress bar lets you know how far you have gotten. In Batch mode see the DEZING option.
- **Smooth:** This button, if clicked, will locally smooth the image, by replacing each pixel with an average including that pixel and the surrounding pixels. This may help in identifying interesting features in a noisy image. It should not ordinarily have a big effect on line plots, but the process does actually throw away some information, so you should probably not do it before making publication-quality plots. Note that this process is irreversible, although the original data file is untouched. If you are unhappy with the results, you will have to read in the data again. For most applications the Condense option may in many cases be closer to what you want. The smoothing process can take several seconds; a progress bar lets you know how far you have gotten. In Batch mode see the SMOOTH option.



- **Condense:** This button, if clicked, condenses the image by summing together multiple pixels. The drop-down menu above it specifies how many pixels are to be combined--for example, if 3x3 is selected then each region of 3x3=9 pixels is summed into one pixel, resulting in a dataset that is 1/9 the size of the original with each pixel containing on the order of 9 times the original value. Like the Smooth option, this may help in identifying interesting features in a noisy image. However, it does a better job than Smooth on low statistics data, since the data are summed rather than averaged so that no counts are thrown away. It also has the effect of speeding up subsequent calculations, since the size of the dataset over which calculations are done is substantially reduced. The net effect is as if you had a detector with lower spatial resolution (larger pixels). If you are unhappy with the results, you will have to read in the data again. The condensing process can take several seconds; a progress bar lets you know how far you have gotten. In Batch mode see the CONDENSE option.



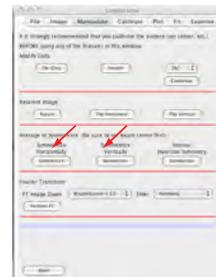
- **Rotate:** This button, if clicked, will rotate the image 90 degrees counterclockwise. The rotation process can take several seconds; a progress bar lets you know how far you have gotten. In Batch mode see the ROTATE option.



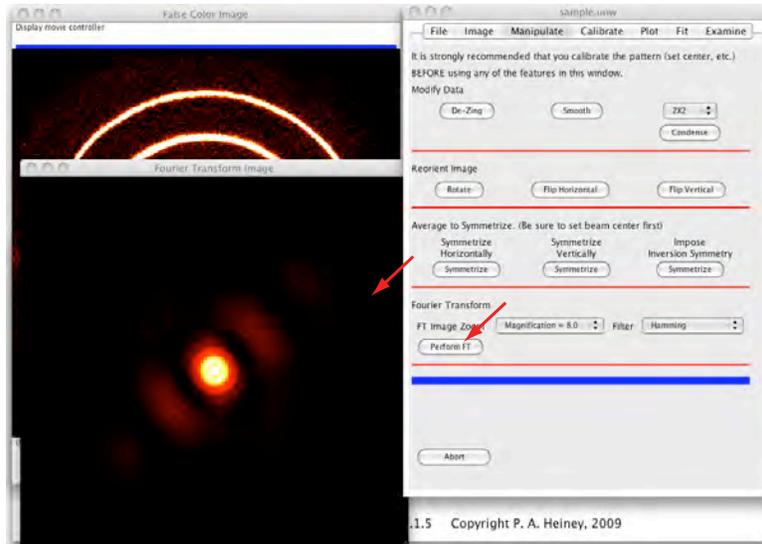
- **Flip Horizontal:** This button, if clicked, will flip the image about a vertical axis. The reflection process can take several seconds; a progress bar lets you know how far you have gotten. In Batch mode see the FLIPHORIZONTAL option.
- **Flip Vertical:** This button, if clicked, will flip the image about a horizontal axis. The reflection process can take several seconds; a progress bar lets you know how far you



- have gotten. In Batch mode see the FLIPVERTICAL option.
- **Symmetrize Horizontally:** This button, if clicked, will “fold” the data about a vertical axis passing through the center position so that the right and left sides are symmetric. Note that it is important to carefully set the beam center before doing this. Note also that data size usually increases somewhat, with the new center forced to be in the x-pixel at the center of the image and a concomitant adjustment of the Q-range calibration parameter. In Batch mode see the HORIZSYM option.



- **Symmetrize Vertically:** This button, if clicked, will “fold” the data about a horizontal axis passing through the center position so that the top and bottom sides are symmetric. Note that it is important to carefully set the beam center



In the FFT algorithm, an array of N real elements is replaced by a new array of N complex elements. The smallest element corresponds to a periodicity of 2π over the largest element in the original array. Specifically, in the Datasqueeze implementation the following operations are performed:

1. If needed, the data are padded with zeros to create a square array whose dimensions are an integer power of 2 (1024, 2048, etc.).
2. Data excluded by the user-defined Mask are replaced with zeros.
3. The data are multiplied by a filtering function whose function is to minimize ringing and aliasing effects.
4. Each row of the array is subjected to the FFT algorithm, replacing the original numbers with new complex numbers.
5. Each column of the array is subjected to the FFT algorithm.
6. The power spectrum is calculated as the sum of the squares of the real and imaginary parts.
7. The array is folded onto itself so that the zero (corresponding to the summed values of the original array) is at the center, rather than the corners, of the array.
8. This new power spectrum is displayed in a false color image similar to that originally used to display the data.

Use: To execute the FFT, after reading in the data the user selects the desired Zoom and Filter, then clicks on the Perform FT button in the Manipulate panel.

Magnification: This drop-down menu selects which part of the calculated spectrum will be viewed. Keep in mind that the outer part of the FT image reflects frequencies corresponding to length scales of one pixel, and are probably uninteresting. For a typical diffraction pattern, which fills much of the detector, the user will want to zoom in to near the center of the FT image.

Filter: As discussed above, spurious ringing effects are minimized by filtering the data using a smooth function that goes to zero at the boundaries. At present, there are four possibilities for the prefilter; the addition of more will be driven by user input:

1. **Hamming:** This is probably the most commonly used filter in discrete Fourier transforms. Starting with an array of N intensities, $z[i]$, the data are multiplied by a sine function that goes to zero at the end points:

$$z'[i] = z[i] \sin^2\left(\frac{\pi i}{N-1}\right)$$

The same filter is applied in both the horizontal and vertical directions.

2. **Pseudo-Lorentzian:** There does not seem to be a good name for this function, which is nevertheless commonly used in the numerical analysis community. The data are multiplied by the function:

$$z'[i] = \frac{z[i]}{1 + \left(\frac{4(i - N/2)}{N}\right)^6}$$

This function does not quite go to zero at $i=0$ or $i=N$, but it is small. Compared to the Hamming function, it remains relatively large far from the center of the array.

3. **Gaussian:** The data are multiplied by the function:

$$z'[i] = z[i] e^{-4(i - N/2)/N)^2}$$

4. **None:** Not recommended, this option leaves the data untouched.

Advice: The best results will be obtained for data that have relatively low background (or at least, very good signal:background) and for which both the data and the background are close to zero near the edges of the detector. To improve the quality of the FT image:

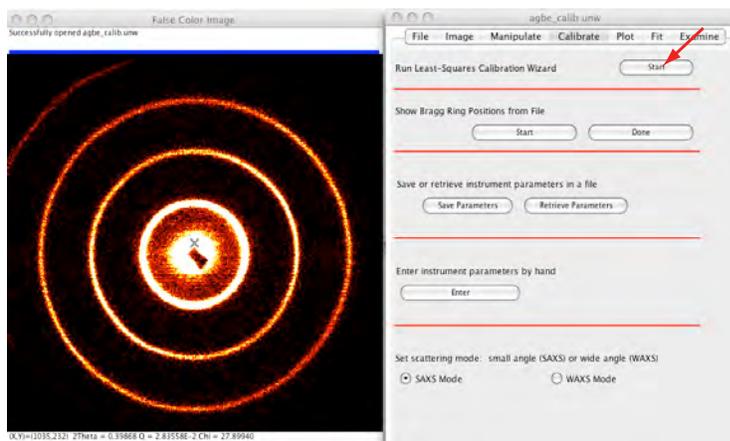
1. If possible, subtract any background when reading in the data.
2. Use the Mask feature to mask out any features not believed to arise from sample scattering. These might include parasitic scattering near beam zero, scattering at wide angles not believed to arise from the sample, etc.
3. Consider using the Vertical, Horizontal, or Inversion symmetrizing features.
4. Unless the interesting features in the data alternate from pixel to pixel, you will probably want to zoom in by a factor of 4 or more. To put it

differently, if the instrumental resolution is N pixels wide, you will want to zoom in by at least a factor of N .

Calibrate Panel

This panel allows you to set the center and scale of the image. There are multiple ways to do this:

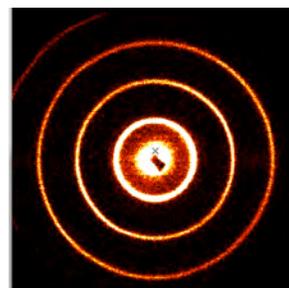
1. **Autocalibration** If you know the d -spacing, q -value, or 2θ of one or more Bragg rings from your calibration sample, then the autocalibration wizard can automatically find the detector parameters (with a little help).
 - To start, click on “Start” underneath “Run Least-Squares Calibration Wizard”. This brings up the calibration wizard, which will lead you through a sequence of steps. At each point you can back up using the “Previous” button.



- The first box asks you to select your instrumental mode by checking the appropriate button.

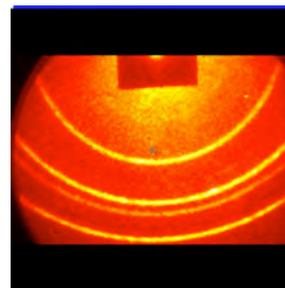


- a. In **Small-Angle** (SAXS) mode, the more commonly-used of the two, the primary beam is on or close to the detector (generally with a beamstop between the beam and the detector), and the face of the detector is approximately normal to the beam. If the detector is relatively close to the sample, the angular range spanned by the detector will be large, while if it is farther away the range will be small, but in either case the widest angle obtainable is generally determined by the size of the detector and the smallest angle by the size of the beamstop. Bragg rings will look

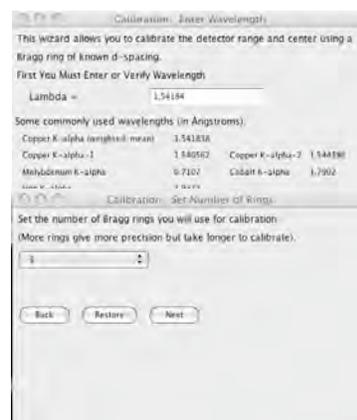


approximately circular on the detector in this case. In Datasqueeze, the relationship between momentum transfer q and pixel number (x,y) can be determined if the instrumental configuration is characterized in terms of the wavelength λ , the q -range (span of the detector in momentum), the coordinates of the pixel where the primary beam hits, and the amount and direction by which the detector is tilted relative to the primary beam (Detector Tilt and Tilt Azimuth).

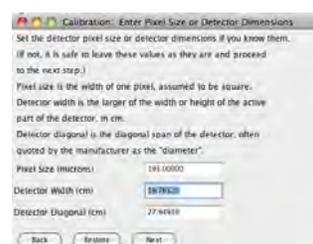
- b. In the less commonly used **Wide Angle** (WAXS) mode, the detector is mounted on a 2θ arm which may be rotated by a wide angle (often 90 degrees or more) from the primary beam. In Datasqueeze, the instrumental configuration is now characterized by the wavelength λ , the q -range (the span in momentum transfer that the detector would have if the beam center were at the center of the detector), the amount and direction by which the detector 2θ arm was rotated relative to the primary beam (Detector 2θ and Detector Azimuth), and the coordinates of the pixel where the primary beam would hit if Detector $2\theta = 0$.



- The next step asks you to verify the wavelength used (in Å). If necessary, retype into the box labeled “Lambda”. A box below reminds you of useful wavelengths for commonly used x-ray targets.
- Now enter the number of Bragg rings that you will use to calibrate the instrument parameters. In SAXS mode the minimum is one ring, but if you do not have an entire circle showing, you will need to use at least two. In WAXS mode, the minimum is two rings. In general, the more rings you use, the better the fit, but also the longer the process takes.

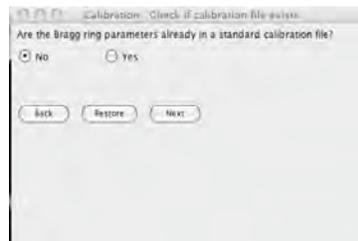


- Next, you are asked to enter in either the pixel size or the detector dimensions. The pixel size is the span of one detector pixel, in micrometers. (The pixels are assumed to be square, which seems to be true for all currently marketed detectors). The detector width is the width or height of the entire pixel array, in centimeters (if the array is non-square, the larger of the two dimensions is chosen). This is the quantity that is referred to as the “diameter” in Batch mode. The detector diagonal is the diagonal span of the pixel array, equal to $\sqrt{2}$ times the detector width, and often quoted by detector manufacturers as the “diameter”. If you do not



know these numbers it is not a problem—just leave them as they are and the calibration wizard will still function properly.

- Next, you are asked if you already have the Bragg ring positions in a standard Bruker calibration file. If you answer “Yes” to the query in this box then you will navigate to the file you want. These files are quite simple, and you can create one yourself using most standard text editors with the provided file (agbe.std) as a template. The first line is a title, and the each remaining line consists of a d-spacing (in Angstroms) and an intensity (which is not used).



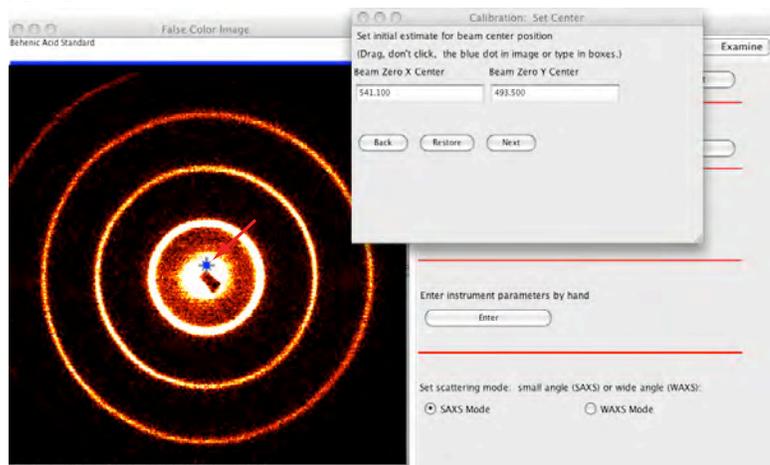
Note however that this is one place where the number format is not locale-dependent; the d-spacings should all use the American format. That is, a d-spacing that is a little over 45 Å should be written as 45.3 rather than 45,3. If you check the “Yes” button then in the next step you will navigate to the calibration file.

Behenic Acid Standard	d-spacing (Å)	Intensity
	58.38	100
	29.19	100
	19.46	100
	14.59	100
	11.676	100
	9.730	100
	8.340	100
	7.298	100
	6.487	100
	5.838	100
	5.387	100
	4.865	100
	4.491	100
	4.170	100
	3.892	100

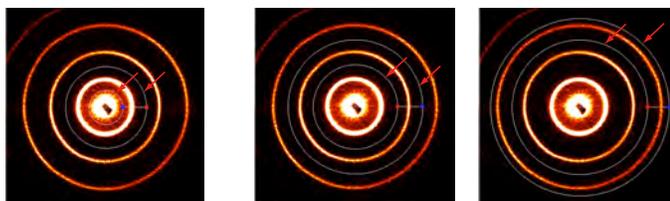
- For the next step in the calibration process, enter the values of momentum transfer q , Bragg angle 2θ , or d -spacing d for each of the rings that you will use for calibration. If you have read in a calibration file, they will already be entered in editable boxes; if not, default values will be entered. For the calibration to work properly you need rings, not sharp peaks from single crystals or fiber diffraction patterns, so your calibration material should be a powder. Type the parameter appropriate for each ring into one of the boxes labeled Q , 2θ , or D .
- The next few steps in the wizard depend on whether you selected SAXS or WAXS Mode. **For SAXS mode:**
 - Set the approximate position of the beam center. (This is the position where the direct beam would hit the detector; it is usually hidden by a beam stop so you have to make an estimate). You can do this by dragging the beam center position on the false color image, or by



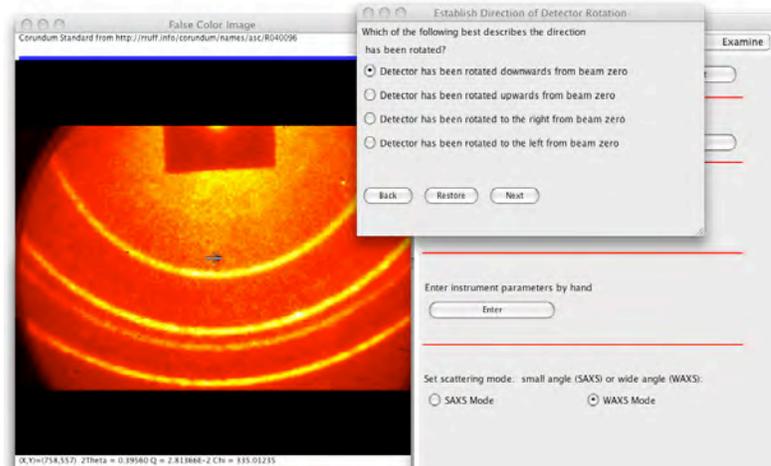
typing in the “X-Center” and “Y-Center” boxes.



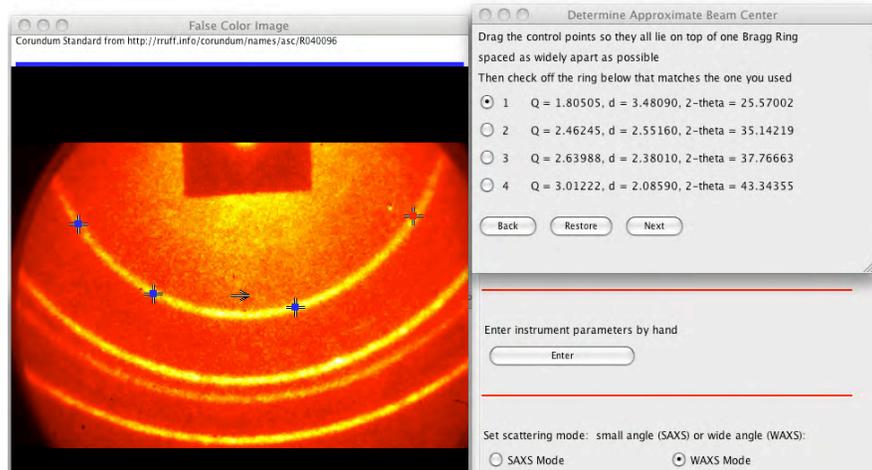
- For each ring, you will identify the feature in the data corresponding to the ring that will be analyzed. Drag the blue dots on the false color image so that the inner and outer rings span the Bragg ring of interest, just as if you were going to make a q-chi plot of that ring. This will tell Datasqueeze over what region it should do the least-squares fit. You want to make sure that the inner ring is everywhere at smaller radius, and the outer ring everywhere at larger radius, than the feature of interest, and also that no other strong features are included within that region.



- **For WAXS Mode:**
 - Datasqueeze first attempts to establish the general orientation of the detector relative to the beam zero. For a laboratory apparatus, the detector has quite likely been rotated to the right or left in a horizontal plane (so that the beam zero is to the left or right of the detector). For a synchrotron facility, the detector has likely been rotated in a vertical direction. Check the box that best corresponds to the direction the 2theta arm has moved.



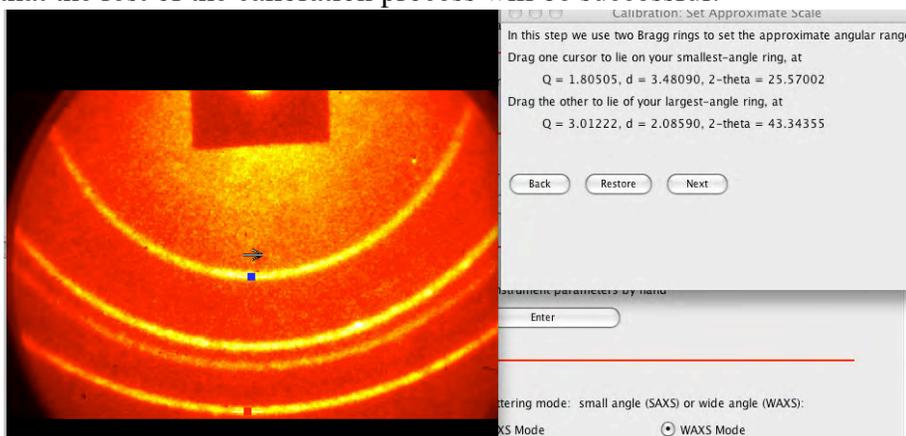
- In the next step, Datasqueeze uses the curvature of one of the Bragg rings to determine the approximate position of the beam center position (which may or may not be on the detector). Four blue control dots are shown. Drag them so that they all lie on top of one of the Bragg rings in the image, as widely spaced as possible. In the wizard box, be sure to correctly identify which ring you used.



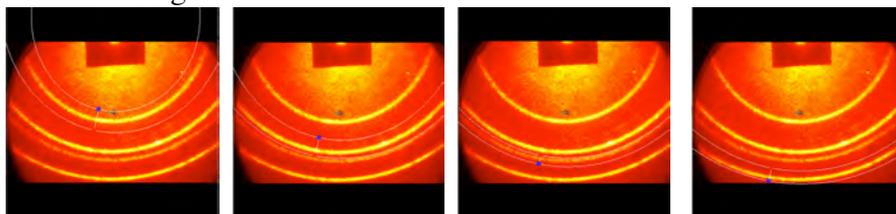
- In Wide Angle mode we assume that the detector is mounted on a 2theta arm which has been rotated away from the primary beam. You are asked to enter the angle in degrees by which the detector has been rotated away from the beam zero. This will probably be the “2θ” value in whatever software you are using to control the sample goniometer.



- Before optimizing parameters, we need a good estimate for the angular width of the detector. This is accomplished by marking the positions of two of the Bragg rings that you will use for calibration. You should see on the false color image two blue cursor dots. Drag one of these cursors to the lowest-angle ring that you will use for calibration and the other to the highest-angle ring. **Important:** These must be the lowest- and highest-angle rings that you are actually going to use for calibration, not other rings that might be on the image and not intermediate-angle rings. If this step is not done correctly, it is unlikely that the rest of the calibration process will be successful.



- For each ring, you will now identify the feature in the data corresponding to the ring that will be analyzed. Drag the blue dots on the false color image so that the inner and outer rings span the Bragg ring of interest, just as if you were going to make a q-chi plot of that ring. This will tell Datasqueeze over what region it should do the least-squares fit. You want to make sure that the inner ring is everywhere at smaller radius, and the outer ring everywhere at larger radius, than the feature of interest, and also that no other strong features are included within that region.

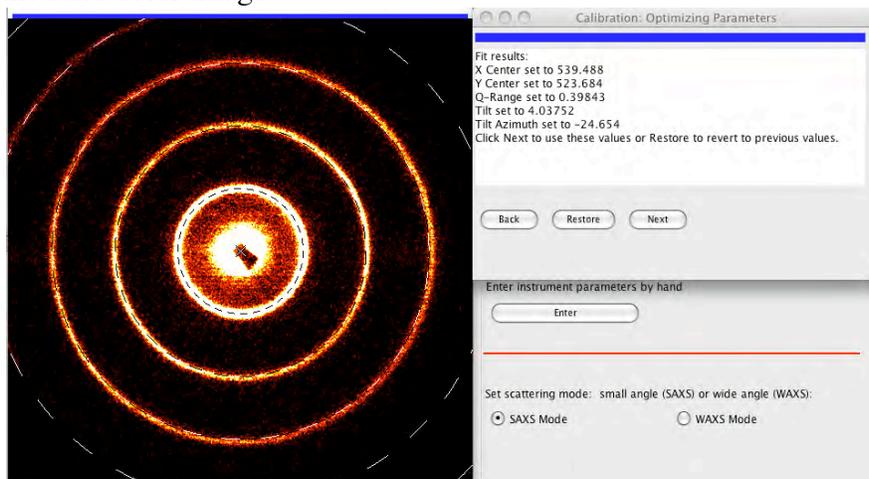


- In the next step (for either SAXS or WAXS mode) you are asked to confirm the starting values of the calibration parameters before Datasqueeze enters the least-squares-fit phase of the optimization. The box displays the starting values of the parameters that will be used in optimizing the detector calibration. In most cases, you will probably wish to leave them untouched and proceed to the next step. However, if you feel that you have a better idea of what a particular parameter should be, you can



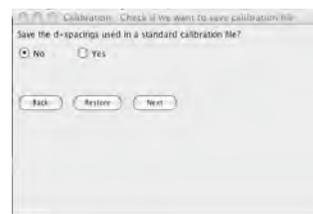
enter that value in the appropriate box. You can also decide not to optimize a particular parameter by checking the box next to that parameter. This is indicated if you are certain that you already know the best value for that parameter, or if you find that in the next step Datasqueeze is unable to accurately optimize that parameter, or if you want to hold that parameter fixed to maintain consistency with some previous data analysis.

- Datasqueeze will now attempt to refine the detector parameters. Depending on the size of your image, and the reflection chosen, this may take some time. After the calibration process is finished, the optimized parameters will be printed in the box and also the calculated ring positions will be displayed in the false color image.



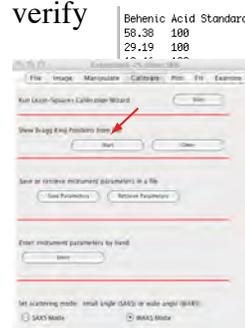
If they look reasonable to you, and the rings on the false color image look centered, click “Next.” If you wish you can cut-and-paste the information to another document. If the parameters appear to be physically unreasonable, or if the calculated rings are visibly inconsistent with the measured rings, you will wish to redo the calculation (click “Back”). It is possible that you made a mistake in one of the earlier steps (for example, in WAXS mode, incorrectly identifying the ring that you used for estimating the beam center position). It is also possible that one or more of the fitting parameters need to be checked (or unchecked) in the previous step to change which parameters are allowed to vary.

- You are given the opportunity to save the d-spacing(s) you used in a calibration file. If you already got the parameters out of a *.std file there is probably no point to this, but if you typed them in by hand then it may be convenient to save them for future use.
- After the best calibration parameters (wavelength, q-range, tilt, etc.) have been determined, they can be saved in a file for subsequent use. Previously saved parameters can be retrieved in Batch mode using the “RETRIEVEINSTRUMENTPARAMETERS”



command or in user mode using the Retrieve Parameters button in the Calibration window.

2. **Show Bragg Ring Positions from File:** You can visually verify your alignment by comparing the positions and radii of measured Bragg rings with those calculated using a standard Bruker calibration file. These files are quite simple, and you can create one yourself using most standard text editors with the provided file (agbe.std) as a template. The first line is a title, and the each remaining line consists of a d-spacing (in Angstroms) and an intensity (which is not used). Note however that this is one place where the number format is not locale-dependent; the d-spacings should all use the American format. That is, a d-spacing that is a little over 45 Angstroms should be written as 45.3 rather than 45,3. This is often performed in conjunction with the “Enter Instrument Parameters by Hand” option (see below).



- To begin, click on “Start” under “Show Bragg Positions from File”. This will open a dialog box which will allow you to navigate through the file system and find the calibration file you want. It should have a .std extension, although this may be invisible under Windows. Select the file you want to use and click Open. A set of circles should appear on the data image.
- Now click “Enter” under Enter Instrument Parameters by Hand, and adjust the center, range, and other parameters until the calculated rings lie on top of the measured rings and the Bragg rings are as circular as possible. **Note:** This can fail if you are not using a calibration file appropriate for the data set you are using. For example, if the largest d-spacing of the calibration sample chosen is still smaller than the minimum d-spacing accessible for the camera length of your apparatus, no rings will appear on your image. Likewise, if the d-spacings are too small, you will get a large number of useless rings at very small radius. Also, you want to make sure that you have set the color scale such that the diffraction rings of interest are actually visible before starting the calibration process.
- Adjust the center positions, detector tilt, azimuth, and Q-range until the calculated circles lie on top of the measured Bragg rings and the Bragg rings are as circular as possible
- **Important:** Remember to click on the Done button when finished. Otherwise, if you click on the image again (even by accident) you may mess up your plot or cause confusion in other parts of the program.



3. **Save or Retrieve Instrument Parameters:** If you have determined the best calibration parameters for a particular data collection, you can save them for subsequent use by using the Save

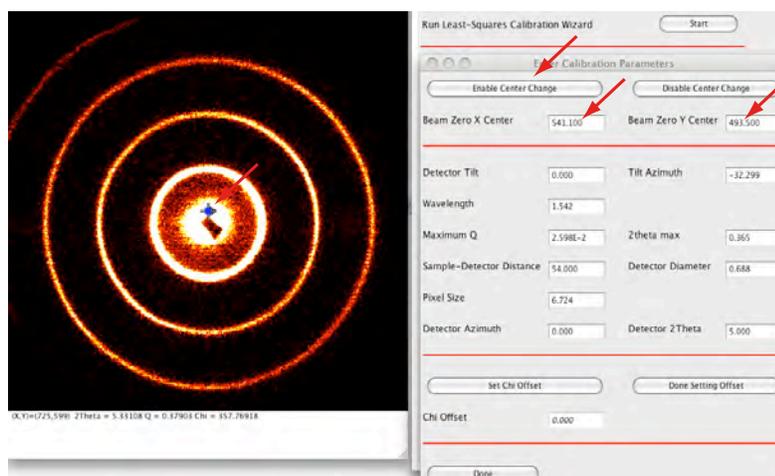


Parameters button to save and the Retrieve Parameters button to retrieve; when running the Least-Squares Calibration Wizard the user is also given the opportunity to save the obtained parameters. Previously saved parameters can be retrieved in Batch mode using the “RETRIEVEINSTRUMENTPARAMETERS” command.

4. **Enter Instrument Parameters by Hand:** If you have previously established the center positions, detector tilt, azimuth, and Q-range, you can also just enter these manually. Click the Enter button to bring up the window for manual calibration parameter entry.



- **Centering** This feature allows you to change the center of the image. In SAXS mode this is the position where the direct beam would have hit if there were no beamstop, and it is indicated by an X on the false color image. In WAXS mode it is the position where the beam would hit if Detector 2Theta were zero, and it is indicated by an arrow pointing in the direction of increasing 2theta. For a detector with N x N pixels, the X-Center and Y-Center values are often somewhere close to (N/2, N/2), although sometimes data are taken in a mode where the beam zero is close to the edge of the detector, and sometimes the beam zero is completely off the detector. In Batch mode see the CEN command. To start recentering, click on “Enable Center Change” (nothing will happen until you do this). The X-Center and Y-Center boxes in the right should change from italic to roman font. At this point, you can change the center either by retyping the desired values in the two boxes, or by dragging the X in the image to the new position.



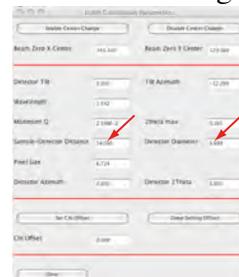
Very important: you must click the “Disable Center Change” button when you are done. Otherwise, mouse clicks to center may still be enabled and you may get confusing results in other parts of the program. The windows will change back to italic when you have done this, and the image will be

recalculated so that the center of the diffraction pattern is at the center of the image (depending on the Zoom option you have chosen).

- Detector Tilt, Tilt Azimuth:** These options are only used in SAXS mode. It commonly happens that the detector is not quite perpendicular to the beam normal. In that case, what should be a circular pattern is elongated perpendicular to the tilt axis. The combination of Det-Tilt and Azimuth correct for this. Det-Tilt is the amount that the detector is tilted (in degrees) and Azimuth is the angle with which the pattern is tilted. For a pattern that appears to be stretched along the x-axis, you should set Azimuth=0 and play with Det-Tilt until the pattern is circular. For a pattern that appears to be stretched along the y-axis, Azimuth should be 90. In fact, any value of Azimuth between -360 and 360 is allowed.
- Wavelength:** This is the wavelength of the radiation used, in Angstroms. The default is the wavelength for Cu-K_α radiation, as is used in most tabletop diffraction units. In Batch mode see the LAMBDA option.

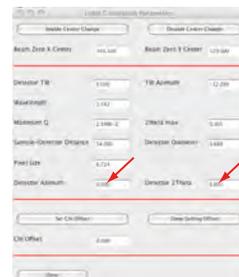


- Maximum Q:** The “Q-range” corresponds to the maximum momentum transfer at one edge (not corner) of the image, assuming that the beam zero is in the center of the image. It is one way of characterizing the angular range of the instrument, the others being “2theta max” and the ratio of “Detector Diameter” and “Sample-Detector Distance”. For small-angle scattering the Q-range will be small while for wide-angle scattering it will be large. 2-theta max and the Sample-Detector distance are automatically updated every time the Q-range is changed, and vice versa. In Batch mode see the Q-RANGE option.
- 2theta-max:** 2theta-max is the scattering angle corresponding to the Q-range; it is the 2-theta at one edge (not corner) of the image assuming that the beam center is in the center of the image. It depends on the camera length. For small-angle scattering 2theta-max will be small while for wide-angle scattering it will be large. The Q-range and Sample-Detector Distance are automatically updated every time 2theta-max is changed, and vice versa. In Batch mode see the TWOTHETARANGE option.
- Sample-Detector Distance:** This is the distance from the sample to the camera, in centimeters. 2-theta max is the arctangent of this distance divided by the radius (Detector Diameter / 2). The Q-range and 2-theta max

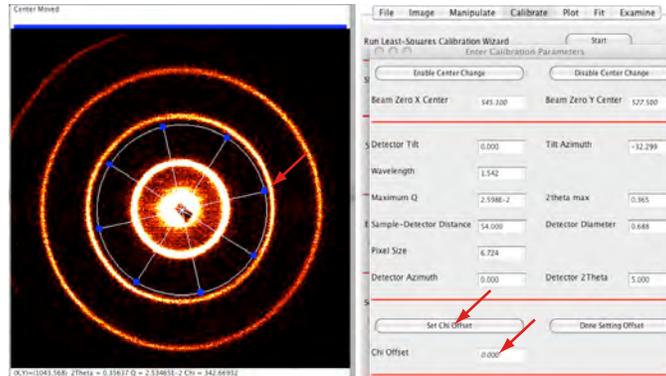


are automatically updated every time Sample-Det is changed, and vice versa. In Batch mode see the LENGTH option.

- **Detector Width:** The detector width is the width or height of the entire pixel array, in centimeters (if the array is non-square, the larger of the two dimensions is chosen). The Q-range and 2-theta max are automatically updated every time the detector width is changed, and vice versa. In Batch mode see the DIAMETER option.
- **Pixel Size:** The pixel size is the span of one detector pixel, in micrometers. The horizontal and vertical dimensions are assumed to be the same. The detector width, and from that the q-range and 2-theta max are automatically updated every time the pixel size is changed. In Batch mode see the PIXELSIZE option.
- **Detector Diagonal:** The detector diagonal is the diagonal span of the pixel array, equal to $\sqrt{2}$ times the detector width, and often quoted by detector manufacturers as the “diameter”. The Q-range and 2-theta max are automatically updated every time the detector width is changed, and vice versa.
- **Detector Azimuth, Detector 2Theta:** These options are only used in Wide Angle model. We assume that the detector is mounted on a 2theta arm which has been rotated away from the primary beam. The Detector Azimuth establishes the direction that the 2theta arm has been rotated; an azimuth of 0 or 180 corresponds to a rotation in the horizontal plane (common to laboratory tabletop units) and an azimuth of 90 or 270 corresponds to a rotation in the vertical plan (often used in synchrotron diffraction systems). The Detector 2Theta is the amount by which the detector has been rotated away from the beam center.
- **Chi Offset:** Chi is the azimuthal angle that defines the orientation of a feature in an image. By default, chi=0 is defined to be in the usual place, at the bottom of the first quadrant. This is fine for powder diffraction patterns or other cases where the orientation does not matter. However, suppose you have a fiber that is almost but not quite aligned with the horizontal or vertical axis. In that case you might want to make qx-qy plots, or to define chi=0 to be along the fiber axis. Datasqueeze deals with this by redefining the meaning of “chi=0”. The “chi offset” is the amount in degrees by which the experimental equator differs from the nominal horizontal direction. To set it:
 - Click on Set Chi Offset. A crosshair pattern should appear in the false color image.



- You can either type your best estimate for the chi offset into the box, or drag the control points on the cross hair until a horizontal or vertical line is aligned with an interesting feature in your data.

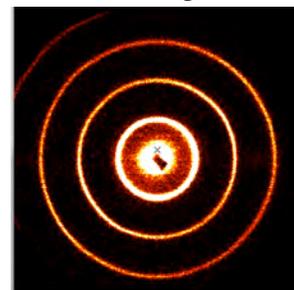


- Click on Done.

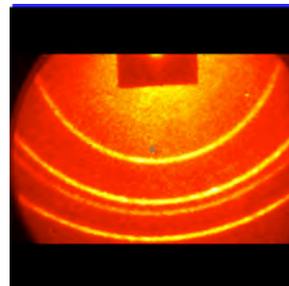
In Batch mode see the CHIOFFSET option.

5. **SAXS Mode:** A pair of radio buttons at the bottom of the Calibrate panel allows you to choose between two calculation modes. X-ray data using area detectors are commonly collected in one of two ways:

- a. In **Small-Angle** (SAXS) mode, the more commonly used of the two, the primary beam is on or close to the detector (generally with a beamstop between the beam and the detector), and the face of the detector is approximately normal to the beam. If the detector is relatively close to the sample, the angular range spanned by the detector will be large, while if it is farther away the range will be small, but in either case the widest angle obtainable is generally determined by the size of the detector and the smallest angle by the size of the beamstop. Bragg rings will look approximately circular on the detector in this case. In Datasqueeze, the relationship between momentum transfer q and pixel number (x,y) can be determined if the instrumental configuration is characterized in terms of the wavelength λ , the q -range (span of the detector in momentum units), the coordinates of the pixel where the primary beam hits, and the amount and direction by which the detector is tilted relative to the primary beam (Detector Tilt and Tilt Azimuth).



- b. In the less commonly used **Wide Angle (WAXS)** mode, the detector is mounted on a 2theta arm which may be rotated by a wide angle (often 90 degrees or more) from the primary beam. In Datasqueeze, the instrumental configuration is now characterized by the wavelength λ , the q-range (the span in momentum transfer that the detector would have if the beam center were at the center of the detector), the amount and direction by which the detector 2theta arm was rotated relative to the primary beam (Detector 2Theta and Detector Azimuth), and the coordinates of the pixel where the primary beam would hit if Detector 2Theta = 0.



In Batch mode see the WAXSMODE option.

Plot Panel

This panel allows you to change decide how a linear (x - y) plot will be made, and to launch the plot. The x - y data generated can be saved in ASCII form; see the FILE Menu. To use these features in Batch mode see the PLOT, AXISRANGE, CHIOFFSET, SELECTREGION, SETANTIALIAS, SETINTENSITYEXPONENT, and SETMARKERINDEX commands.

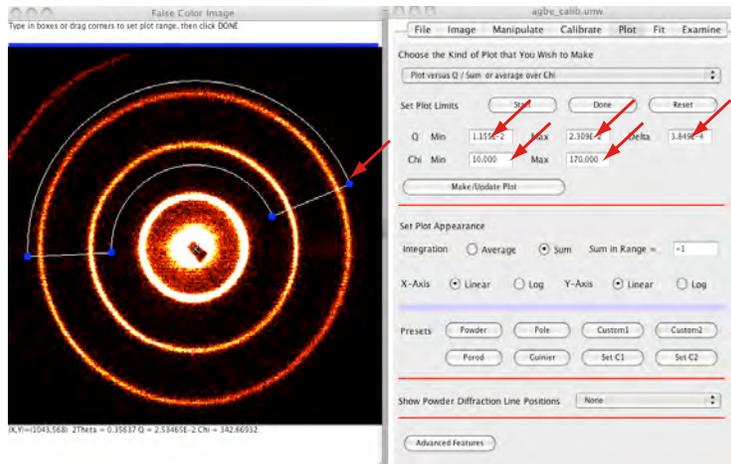
- Independent/Dependent Variable** Since we are converting a two-dimensional data array into a one-dimensional plot, the data have to be collapsed in some way. The possible variables are Q (the magnitude of the momentum transfer), Q^2 (the momentum transfer squared), 2-theta (the scattering angle, in degrees), Chi (the azimuthal angle, in degrees), Q_x (the horizontal component of the momentum transfer), Q_y (the vertical component of the momentum transfer), Ellipse (used for azimuthal plots where the integration is performed going around an elliptical rather than a circular path), or the fraction of the distance along an arbitrarily chosen line segment. A drop-down menu allows you to choose which will be the plotting parameter and which will be the parameter over which we sum or average. For example, selecting *Plot versus Q / Sum or average over Chi* causes the data to be plotted versus Q (a radial plot) while integrating over the azimuthal angle. Selecting *Plot versus Q^2 / Sum or average over Chi* together with a logarithmic y axis (described below) produces a Guinier plot of $\log(\text{intensity})$ versus Q^2 . Selecting *Plot versus Chi / Sum or average over Q* gives a polar plot of intensity versus azimuth at constant radius, while *Plot versus Chi / Sum or average over Q-Ellipse* produces a plot of azimuthal intensity traced over an elliptical path. (Normally, you probably want a circular path; the elliptical path might be useful in cases of a distorted scattering pattern resulting from a strained sample). Note that the scales of Q and 2-theta



depend on having first properly calibrated both the center and the scale of the image, using for example the Calibrate panel. For most variables, when the plot is made the independent variable will be the actual value of Q, Chi, or whatever. For Line plots, there does not seem to be an obvious choice for what we mean by the independent variable, so the intensity is plotted versus fraction of distance along the chosen line segment (0 to 1).

- **Set Plot Limits** This portion of the panel allows you to select the region over which you will make the plot.

- First note that you need to have opened a data file for anything in this section to work.
- Click on Start to begin; nothing will happen until you do this.
- Once you have “started” some curves should appear on the image to the left and some new text boxes should appear below the Start button. If you selected your independent/dependent variables to be Qx/Qty or Qy/Qx you will get a rectangle. If you chose Line you will get a line segment. If the dependent variable is Ellipse you will get an ellipse. Otherwise you will get a pie-shaped wedge.
- At this point, you can change the region of integration either by typing in the appropriate boxes or by dragging on the corners of the box or pie shape in the image. To drag on the corner, click on the large dot at one of the corners, wait until the color changes from blue to red, and then drag to the new position. If all of the limits are outside of the range of the image, you might not see any lines or corners at all. In that case you should type values in the appropriate boxes until you get the region of interest visible on the false color image.



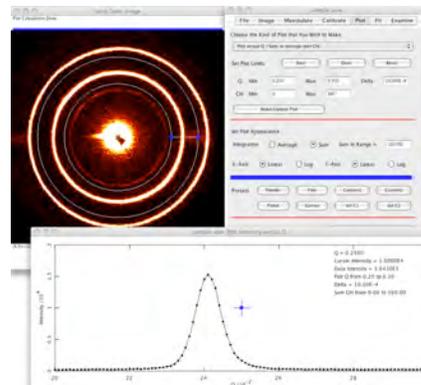
- The limits for the Line and Ellipse options behave somewhat differently from the others. For the Line option, the “Low” and “High” X- and Y-Pixel values are the coordinates of the two ends of the line segment. “Fract” is the equivalent of “Delta” for the other plots—the number of

points will be $1 + 1/\text{Fract}$. Pixels are included in the plot if they are within $\text{Width}/2$ of the line segment itself. So a Width of zero will in general not include any pixels at all. (The default is 3). It is suggested that for this kind of plot you will almost certainly want to Average rather than Sum the data. For the Ellipse option, the independent variable is always Chi and always goes from 0 to 360°; the dependent variables are the lengths of the semimajor and semiminor axes a and b (with however b being allowed to be greater than, equal to, or less than a).

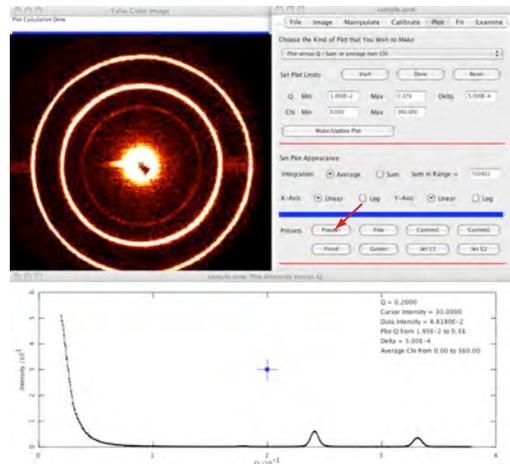
- You also need to set the “delta”—the increment over which the independent variable is plotted. If you make it too large you will have a plot with poor resolution. If you make it too small you will end up with a noisy plot. (If you make it smaller than one pixel you will get a very strange-looking plot.) Also, there is a defined limit on the maximum number of data points, currently set at 2000. Note that if you are going to make plots of a number of related data sets, you probably want to set the integration region and the delta to be the same for each data set.
- Reset button: if you have somehow set parameters to values that are so inconsistent with the data set that no control buttons appear on the screen, hitting the Reset button will restore the starting parameters to something sensible.



- **Make/Update Plot:** When you have chosen the type of plot and the plot limits, click this button to actually generate the x-y plot. (The plot is not automatically refreshed every time you change something like the integration range; you have to do this manually). This may take several seconds if you have selected a new data region, but will go very fast if you have just changed a parameter such as linear vs. logarithmic axes.



- **Integrate/Average** You have two choices here.
 - If you choose “Sum” then the program will integrate over the dependent variable. That is, for each value of the independent variable (say, Q) it will add up the values of all pixels between Q and Q + delta Q and inside the range of the dependent variable (say, Chi) chosen.
 - If you choose “Average” then the program will plot the average counts/pixel over the same data range. This gives something more closely resembling the results of a measurement with a one-dimensional detector (e.g., a triple-axis diffractometer) in which a $\sin(\theta) \sin(2\theta)$ Lorentz factor is often applied to the data, as described in Advanced Plot Options.
 - In most cases the Average option is probably preferred, since it tends to reduce the non-statistical noise due to pixilation errors.
- **Linear/Logarithmic Axes** For both the x (independent variable) and y (intensity) axes in the plot you can choose a linear or logarithmic scale by clicking on the appropriate button.
- **Summed Counts** The “Sum in Range” box displays the total number of counts in the selected region. It is only updated when the “Make Plot” button is clicked. Editing this box has no effect; it is purely informative.
- **Presets** Datasqueeze provides a number of pre-defined settings for making plots. If you click on one of these buttons everything is done for you, with (almost) no decisions to make. Most of these settings, except for the Custom Presets, are useful only in SAXS mode.
 - **Powder Plot** This makes a plot similar to what you could get in a traditional θ - 2θ powder diffraction scan. It is equivalent to setting the Independent/Dependent variables to Q/Chi, so that we plot versus $Q=2\pi/d$ and integrate over the azimuthal angle. The scale is set to run from 0.05 to 0.95 of the Q-range. The increment is set to a round number such that there are something close to 500 points in the plot.



- **Pole Figure Plot** This plots the intensity versus azimuth angle (chi). It is equivalent to setting the Independent/Dependent variables to Chi/Q, so that we plot the azimuthal angle and integrate over $Q=2\pi/d$. Q is set to average from 0.05 to 0.95 of the Q-range. Chi runs from 0 to 360 degrees in steps of 2 degrees.



- **Guinier Plot** The Guinier law states that in the limit as q goes to zero the scattered intensity is proportional to $\exp(-q^2 R_g^2/3)$ where R_g is the radius of gyration of the scattering object. Accordingly, if we plot $\log_{10}(I)$ versus q^2 the slope can be related to the radius of gyration. Clicking the Guinier button is equivalent to setting the Independent/Dependent variables to Q^2/Chi , so that we plot versus Q^2 and integrate over the azimuthal angle. The scale is set to run from $Q^2 = (0.05 \times Q\text{-range})^2$ to $Q^2 = (0.95 \times Q\text{-range})^2$. The increment is set to a round number such that there are something close to 500 points in the plot. A logarithmic scale is used for the Y-Axis.



- **Porod Plot** The Porod law states that, for small-angle scattering from isolated objects (spheres, rods, sheets, etc.), as q becomes large the scattering is proportional to $1/q^4$. Accordingly, a plot of $q^4 I(q)$ should approach a constant. Clicking the Porod Plot button is similar to clicking the Powder Plot button except that n is set to 4 in the Multiply by X^n box so that we plot $q^4 I(q)$. Note that for most real data this plot will *not* approach a constant. This can happen for two reasons. First, there may be real wide-angle scattering due to interparticle correlations. For example, in a polymer melt there will be wide-angle scattering due to chain-chain correlations. Second, there may be a background signal due to air scattering, scattering from windows, etc. If this background is constant it will lead to a rising curve in a Porod plot. If you believe that this is happening you may wish to play with subtracting a constant number of counts per pixel in the File Panel.

- **Custom Presets:** You can save two different settings for user-defined plot parameters. This is useful if you want to plot a number of different data sets in exactly the same way, even in different multiple analysis sessions. Suppose, for example, that you want to define “Custom Preset 1.” You do this in the following way:



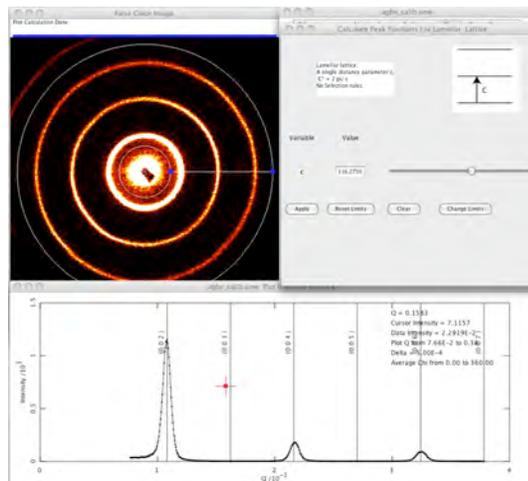
- Using the other plot controls, set up a plot exactly the way you want it (correct Dependent/Independent variables, right choice of axis log/linear buttons, right choice of plot range and increment, etc.).

- Click the “Set C1” button. Datasqueeze will now remember these settings. If you answer affirmatively to the “save current settings” query when you exit, it will even remember these settings from one session to the next.
- Now when you click “Custom1” you will get the same kind of plot even if you have done other plots in the interim.

• **Show Powder Diffraction Lines:** This feature can be helpful in indexing powder diffraction patterns of unknown symmetry. It will not replace a full Rietveld structural refinement, but it can be used to get a preliminary idea of the symmetry of the underlying lattice. After doing a radial plot of some sort (plotting versus Q , 2θ , or Q^2), set the drop-down menu under “Show Powder Diffraction Line Positions” to select the symmetry of a lattice to test. (Possibilities are a lamellar structure, which is one-dimensional, 6 two-dimensional nets, or the 14 three-dimensional Bravais lattices). A new window will appear.



- At the top of the window are some basic information and a simple schematic relating to the lattice selected. Below this are text windows and sliders for the parameters that can be varied for this lattice. This will include some or all of the lattice parameters, a , b , and c , and the lattice angles α , β , and γ . In general, only some of these parameters can be varied. For example, in a rhombohedral lattice, $b=a$, so b cannot be varied, and there is one lattice angle γ that can be varied.



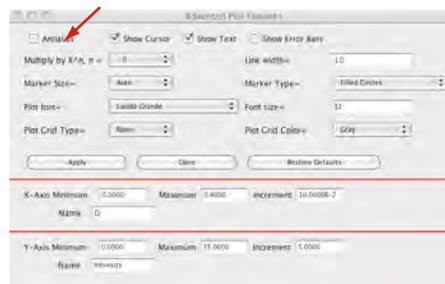
- Click “Refresh” to show the indexed diffraction line positions on your plot.
- Type into the boxes or drag the sliders to change the variables.
- Click “Reset Limits” to reset the range of the sliders from the plots— Datasqueeze does its best to guess a reasonable range for each parameter given the limits of the plot and the symmetry of the lattice.
- Click “Clear” to remove the diffraction lines from the plot.

- Click “Change Limits” to pull up yet another window for manually setting the range of each slider.

Note that if the “Show Powder Rings”; check box has been set in the Image panel then rings for the region of interest will also appear on the false color image.

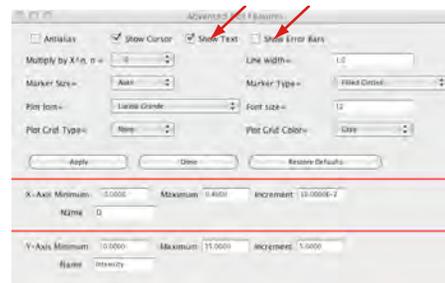
- **Advanced Plot Features:** Clicking the “Advanced Features” button brings up a new window with less-often-used plot features. For most of these, it is necessary to click Apply after selecting the desired option. In detail, the options available in the Advanced Features window include:

- **Antialiasing** When making a plot where there are a relatively small number of pixels per data point, “pixelation” noise can be significant. By checking the “Antialias” box, you can correct for this to some extent. The program breaks each pixel up into 4 sub-pixels, each with 25% of the intensity, and apportions the intensity accordingly. This results in cleaner looking plots, but also takes approximately 4 times as long to calculate, so it should only be turned on when you are satisfied that you have the plot range, etc., set the way you want. Implemented in Batch mode using SETANTIALIAS.



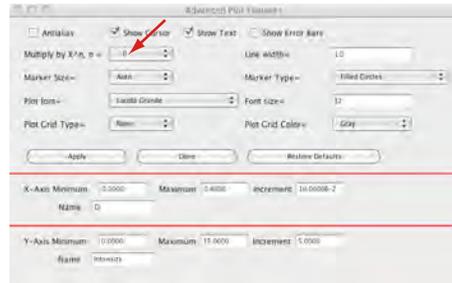
- **Show Cursor:** By default, the plot display contains a cursor which, when dragged, allows the user to obtain a text reading of the independent and dependent values corresponding to the cursor position. This cursor, and the corresponding text in the upper right hand corner of the plot, can be disabled by unchecking the “Show Cursor” box.

- **Show Text:** The upper right hand corner of a plot normally shows information on how the plot was constructed, as well as dynamically updated information on the current cursor position. This text display can be disabled by unchecking the “Show Text” box.



- **Show Error Bars:** If the “Show Error Bars” box is checked then error bars are shown on the plot. This is only recommended if a relatively small number of data points are displayed. Implemented in Batch mode with the SHOWEB command.

- **Multiply by x^n :** Normally one wishes to plot the intensity versus the independent variable. But one can also multiply by a power of the independent variable. For example, to make a plot of Q^2I versus Q , select Q/χ button and set $n=2$ in the Multiply by X^n box.
- **Divide by $A + B \cos^2(2\theta)$:** For an initially unpolarized beam, the scattered beam intensity is reduced by a factor



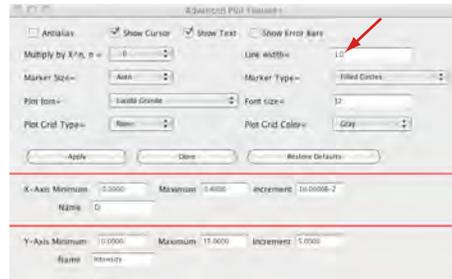
$$I' = I \left(\frac{1 + \cos^2 2\theta}{2} \right).$$

For a beam that is initially polarized (for example, due to a monochromator crystal between the source and sample, or because the source is synchrotron radiation), the polarization correction becomes

$$I' = I(A + B \cos^2 2\theta)$$

where the coefficients 'A' and 'B' depend on the initial polarization of the beam and the azimuthal direction (χ) of the scattered beam relative to the initial polarization. (See, e.g., B. E. Warren, *X-ray Diffraction*, Addison-Wesley, 1969, or other elementary texts on x-ray diffraction). The user can correct for this effect by dividing the measured intensity by the correction factor. Note that the polarization effect is very small for a typical small-angle geometry, in which case one can safely leave A and B at their default values of 1.0 and 0.0; for other geometries the user should think carefully about exactly how to apply this correction since it does in principle depend on χ . Implemented in Batch mode using SETCOSINEAB.

- **Line Width:** The line width is the width of any of the lines in the plot image, in pixels. It can be set to any value between 0.1 and 10.0. The default is 1.0. If larger plots are created (by dragging the corners) or if the image will be cut-and-pasted in another document, a somewhat larger value may be more attractive. Implemented in Batch mode using SETINEWIDTH.
- **Multiply by $\sin(\theta)$, Multiply by $\sin(2\theta)$:** Depending on the scattering geometry, the scattered intensity may be modified by a factor



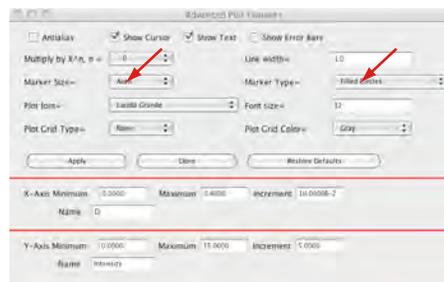
$$I' = \frac{I}{\sin\theta}$$

and/or a factor

$$I' = \frac{I}{\sin 2\theta}$$

For example, when a small single crystal is rotated through the Bragg condition, the correction factor is $1/\sin(2\theta)$, whereas the intensity from a powder sample is $1/\sin(\theta)\sin(2\theta)$, and there is no correction term at all for solution scattering from isolated isotropic particles. A term of the form $1/\sin(\theta)\sin(2\theta)$ is sometimes known as a “Lorentz factor”. (See, e.g., B. E. Warren, *X-ray Diffraction*, Addison-Wesley, 1969, or other elementary texts on x-ray diffraction). The user can correct for this effect by checking one or both of the “Multiply by $\sin(\theta)$ ” and “Multiply by $\sin(2\theta)$ ” boxes in the Advanced Features panel. Implemented in Batch mode using the MULTIPLYSINTHETA and MULTIPLYSINTTHETA options.

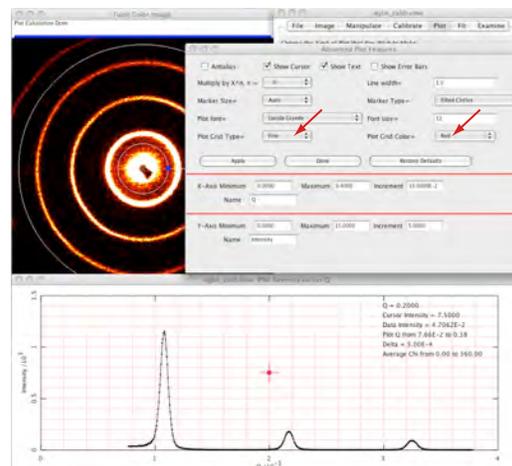
- **Marker Size** A drop-down box allows you to specify the radius of the solid circle that is placed at each data point on the plot. If “Auto” is selected, then the program chooses a radius according to the number of points in the plot. If “Line Only” is selected, then no circles are ever drawn. Otherwise, the user can select any radius from 1.0 to 5.0 pixels in steps of 0.5 pixel. This option is non-volatile, in that it is saved with other preferences and restored the next time the program is restarted by the same user.
- **Marker Type:** A drop-down box allows you to specify the type of symbol used at each data point (filled circle, open circle, etc.). Click the Apply button to implement the change. This option is non-volatile, in that it is saved with other preferences and restored the next time the program is restarted by the same user. Implemented in Batch mode using SETMARKERTYPE.



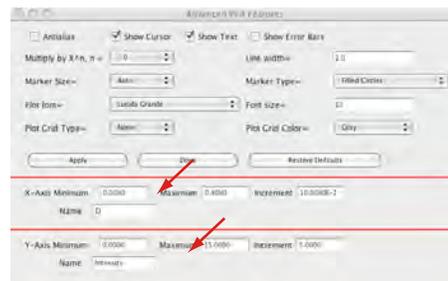
- **Plot Font and Font Size:** A drop-down menu and a text box allow you to change the font and size of labels in the plot. The default size is 12-point. The default font is system-dependent, but is usually Arial. Note that not all fonts will have a pleasing appearance, or even be properly rendered. Implemented in Batch mode using SETPLOTFONT.



- **Plot Grid:** Two drop down menus allow you to overlay a grid on the plot. The first specifies the type of grid: None (default), Coarse (grid lines placed at tick positions), or Fine (grid lines placed at smaller intervals than tick positions). The second specifies the color of the grid (default is black). These are implemented in Batch mode using SETPLOTGRID and SETPLOTGRIDCOLOR.



- **X and Y Axis Options:** After changing the overall plot features and updating the plot with the Apply button, you can change the scale of either the independent or dependent variable, as well as the axis titles. For example, suppose that the plot as generated had the x-axis running from 0.01 to 0.08 in steps of 0.01. Text boxes allow you to reset these values—you could set the lower end of the range to 0.02, the upper end of the range to 0.07, and

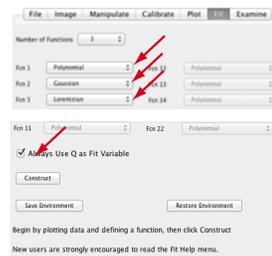


have a tick mark every 0.005. Note that this does not change how the data are calculated, just how they are plotted. Every time the plot is recalculated, the axes, and their titles, are set back to default values.

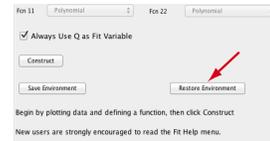
Fit Panel

This panel allows you to set up the model for least-squares fits and then bring up the Fit Parameter Panel from which you control the actual fitting process. A general fitting function, or “model” is the sum of up to 22 sub-functions, each with 3 or 4 independent fitting parameters. Although you will certainly want to play around with least-squares fits on your own, it is strongly suggested that you read the Fit Help documentation later in this manual before doing anything that you plan to show to others! In Batch mode see the CONSTRUCTMODEL, NMODELS, FITVSQ, SETMODEL, SAVEFITENVIRONMENT and RESTOREFITENVIRONMENT options.

- Number of Functions:** This drop-down menu allows you to select how many sub-models you will use, from 1 to 22.
- Function Selection:** Each drop-down menu below the top one allows you to select a particular sub-function.
- Use Q as Fit Variable:** Some functions (e.g. the Rayleigh function for small-angle scattering from spheres) expect Q as the independent variable. If this box is checked, then Q will be used as the independent variable for least-squares fits even if the plot was made versus 2θ or Q^2 . This will be true for all submodels used, even those such as Gaussian which do not necessarily require Q as the independent parameter. If it is not checked, then the variable you used for making the plot will be used even if it is inappropriate for the function employed. If Datasqueeze thinks you might have made a mistake in selecting (or not selecting) this box, a dialog box pops up at the point when you actually perform the fit.
- Construct Button:** When you have your model set up the way you want it, click on this button to bring up the Fit.
- Save Environment:** This option allows you to save everything about your current fit model environment, including the models used and the values of the parameters, in a file for subsequent use. This can be useful if you are fitting many data files to the same model (perhaps quite complicated with many parameters) and you want to start the same place every time. Clicking this button brings up a dialog box that allows you to select a name and location from the file.



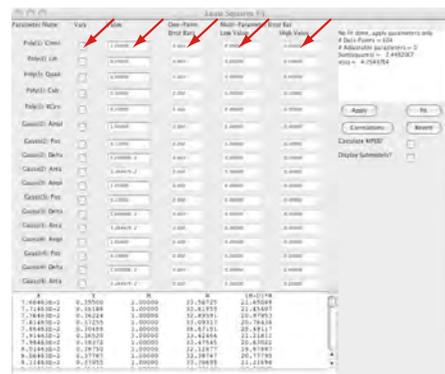
- **Restore Environment:** This option restores the previously saved fit environment. Clicking the button brings up a dialog box that allows you to select the file you want.



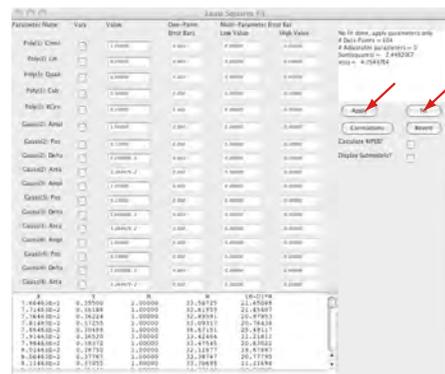
Fit Parameter Panel

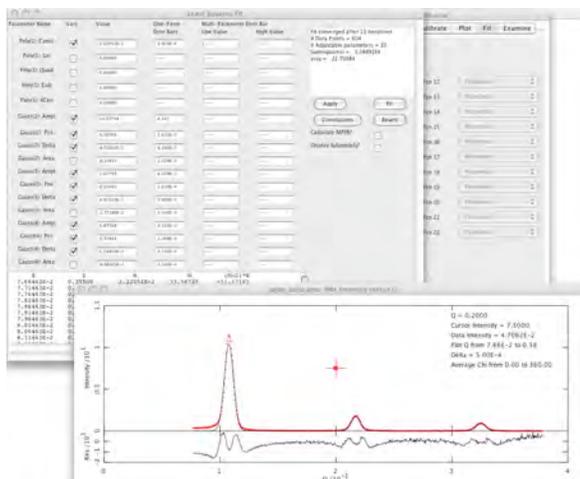
This panel pops up once you have constructed a model and clicked on the Construct button in the Fit Panel. It has the following components (in Batch mode see the SETFITPARAMETER option):

- **Parameter entry:** Each fitting parameter is given a line with some boxes.
 - Next to the name of the parameter is a check box. Check this if you want this parameter varied during the fit; otherwise leave it unchecked.
 - Next is the “Value” of the parameter. Enter an initial guess for the parameter; the final value after the least-squares minimization will also appear here.
 - Next is the one-parameter error bar. This is an estimate of the one-standard-deviation uncertainty in this parameter, assuming that no other parameters were varied.
 - Finally, there are low and high values for the “Multi-parameter error bar.” These are one-standard deviation limits on this parameter assuming that all other variable parameters (those which you have allowed to vary) are adjusted simultaneously.

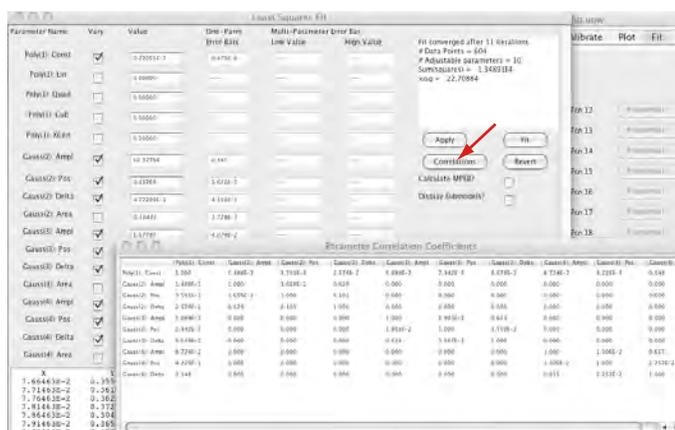


- **Apply Button:** Click here to calculate the model with these starting parameters, not allowing anything to vary. The results of your fit will appear together with the original data in the Line. In Batch mode see the APPLYFIT option.
- **Fit Button:** Click here to perform a least-squares fit, using the starting parameters you entered in the Value box for each parameter, and varying only the checked parameters. The results of your fit will appear together with the original data in the Line. Useful messages may appear in the Fit Message Area of the Fit Panel. The new values of the fitting parameters will appear in the Value boxes, and the estimated uncertainties will appear in the boxes to the right of these. In Batch mode see the DOFIT option.

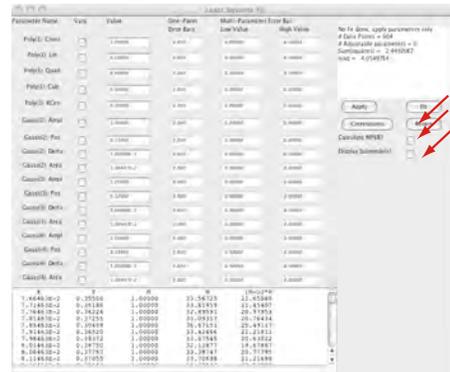




- **Cancel Button:** If a fit is taking a very long time, and you realize that you want to change starting values of parameters or the parameters varied, click here to cancel the fit. Warning: parameters may be left in an indeterminate state at this point.
- **Correlations Button:** Click here to bring up a new window showing the correlations between different parameters. A correlation coefficient of 0 means that two parameters are “orthogonal”—they have totally different effects on the shape of the curve. A correlation coefficient of 1 means that the two parameters are totally correlated, and do exactly the same thing to the model—this is a bad thing. Similarly, a correlation of -1 is bad. You generally bring up this window if a fit fails due to strongly correlated parameters and you want to figure out which ones.



- **Revert Button:** Click here to set all parameters back to the values they had before the last time you performed a fit.
- **Reset Button:** Set all parameters back to reasonable values such that the function is at least likely to display properly on the plot.
- **Calculate MPEB Box:** Check this box to calculate multi-parameter error bars after doing the fit. This slows things down a bit, so you may want to only do this part for the very last fit.
- **Display Submodels Box** Check this box to plot all the individual submodels in addition to the total model.
- **Display Fit Cursors:** Most fit models come equipped with cursors, which allow you to graphically change some or all of the fit parameters. If this box is unchecked, the fit cursors are not shown. (The default is to show all cursors).

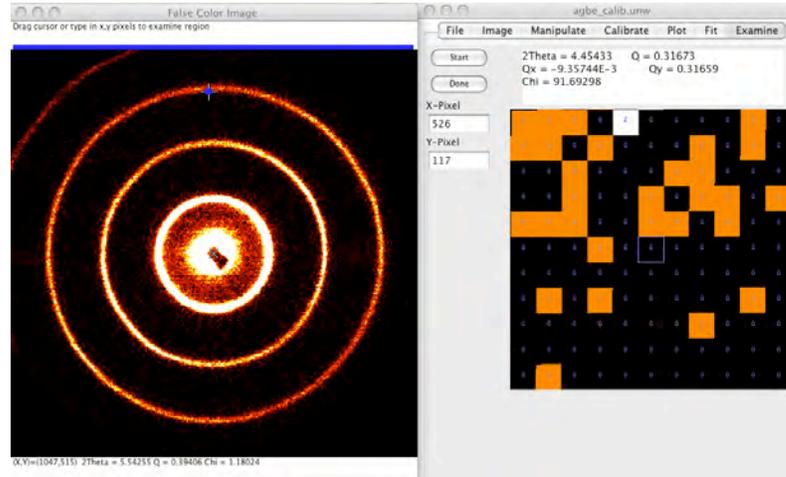


The general procedure for performing a least-squares fit is as follows:

1. Open a data file in the usual way.
2. Make a line plot. The data in the line plot will be what you are fitting to.
3. Use the Fit to construct a model function by specifying the number of sub-functions and the type for each, and click on Construct to bring up the Fit Parameter Panel.
4. Set each parameter to a reasonable starting value.
5. Click the Apply Button and see whether your model with these parameters agrees approximately with the data. If it does not, go back to step 4.
6. Check off the parameters you want to vary, and click on the Fit.
7. If you get unexpected results, you may want to uncheck a few parameters, click on the Revert Button, and try again.
8. Check in the message area to see how well the fit worked, and whether there were any errors. If you are happy with the results, you may want to print the plot showing the agreement between model and data, or save the fitting parameters.

Examine Panel

This panel allows pixel-by-pixel examination of the current image. An 11 x 11 pixel blowup of a selected region appears with the same false color scaling as is set in the Image Panel.

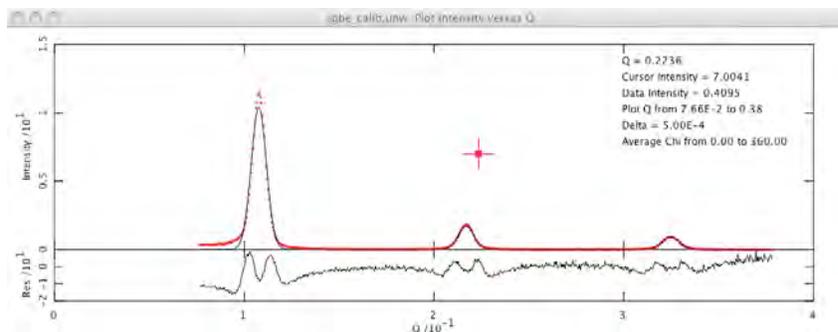


- To start, click the START button. A cursor will appear in the upper-left-hand region of the Image Panel.
- Drag the cursor to the desired position.
- The boxes below the DONE button will show the current X and Y of the selected pixel, which is also the center pixel (outlined in a blue box) in the smaller image to the right.
- The box at the top right continuously updates the current values of Q, Qx, Qy, 2-Theta, and Chi for the center pixel.
- For more accurate positioning, you can type the coordinates of the desired pixel position into the X-pixel and Y-pixel boxes.
- Don't forget to click DONE when you are done, otherwise, future centering or plot selection operations may fail.

Line Plot Image Window

This is not the tabbed Plot, but the white window near the bottom where the plot actually appears. Although considerable control is provided over the appearance of the plot, if you want to manipulate the data in other ways you can save the data in ASCII form using the FILE Menu and import the data into the graphing application of your choice.

- The top part of the plot shows the x-ray data generated from the full data set. If you have done a least-squares fit to the data, it also shows the function calculated with minimized parameters.



- If you have done a least-squares fit, the bottom portion of the plot shows the residual [(model — data) × weight] for each point.
- The line plot is generated by setting all the desired parameters in the Plot Panel and then clicking on the Make Plot button. In addition to the line plot, you will also get a cursor consisting of a horizontal and vertical line with a large dot in the center. You can drag this cursor wherever you want in the window. (Click on the dot, wait until the cursor changes from blue to red, and drag to the chosen position). For further control, click the Advanced Features button in the Plot Panel to bring up a window with more options.
- In the upper right corner you will see three dynamically updated values: the value of the independent variable corresponding to the current horizontal position of the cursor, the value of the intensity that would correspond to the current vertical position of the cursor, and the actual value of the intensity on the curve above or below the cursor. (Linear interpolation is done between adjacent data points). The current values of the independent variable and intensity appear in the upper right hand corner of the window.
- In some previous versions of Datasqueeze, clicking on the x-axis or y-axis legend allowed the user to change the axis scale. This capability has now been moved to the Advanced Plot Features window
- Drag the lower right corner of the plot panel to resize the plot. You may need to re-generate the plot by clicking on the Make Plot button after resizing.

File Menu

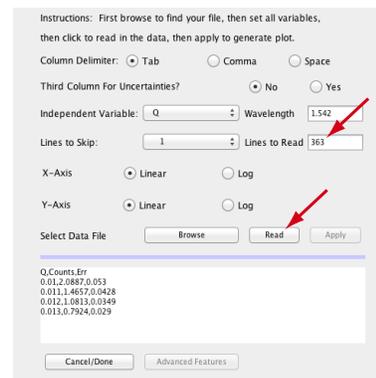
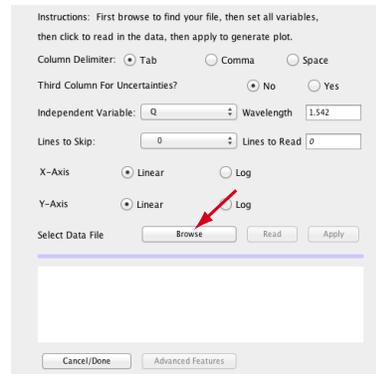
The File Menu is located in the upper-left-hand corner of the screen (Macintosh and Windows standard). There are presently 17 options:

1. **Open Single File:** Opens a single file, with no additive constant, a multiplicative constant of 1, and assuming that you want to add all images in the file if there is more than one. The shortcut key is (ctrl)-O (or Apple-O on Macintosh). If you want to do something more complicated (such as add up multiple files) you should use the File Panel instead. In Batch mode see the READFILE option.
2. **Open Recent Files:** Allows you to re-access a recently opened file, obtained from a list in a drop-down menu. This is opened as a “single file” in the same manner as the previous item.

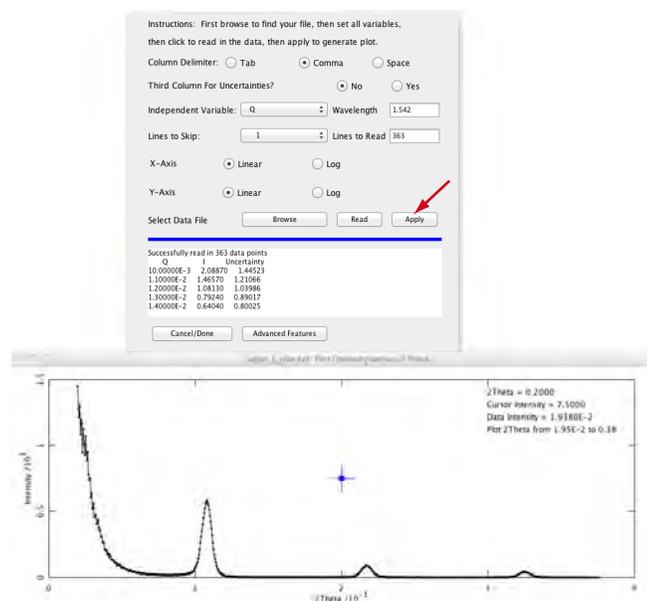
3. **Import 1D Data:** This bypasses the false color screen, and imports a one-dimensional (x-y) data set, which is then plotted. This data set can then be analyzed via least-squares fits or powder indexing. The input file format is that created by the “Save Plot” command, so a common usage would be:

- Open a 2D data file, and create an x-y plot as a 1D cut through the data. (But you might also have used some other application to create x-y data).
- Save the data (Save Plot) in a text file
- Open the text file with an external editor or spread sheet and modify it. For example, you might want to remove “bad points” or stitch together multiple plots from different angular ranges.
- Import the modified file using Import 1D Data for further file analysis in Datasqueeze.

- Your data file should be in ascii format. Each line should minimally consist of an independent variable (e.g., Q) and the measured intensity, with an optional third column for uncertainty in the intensity, all separated by spaces, commas, or tabs. When the menu item is activated, you get a new window as shown below. Buttons in the top row select the column delimiters. The second row of buttons allows you to state whether there is a third column for uncertainties. The drop-down “Independent Variable” box allows you to choose the name of the independent variable. If you have some header information at the top of your file, select an appropriate number under “Lines to Skip” so that the first line read in is actual data. You can choose whether the X and Y axes should be plotted on linear or log scales. (The data in the file itself should always be on a linear, or absolute scale). You can also redefine the wavelength that was used for this data set. Start by clicking on Browse to choose an input file. When this is done, you will either get an error message or the first few lines of the file in white box. After looking at the file, you may wish to adjust the Column Delimiter and other settings above. You can also change the number of lines to read in (box next to Lines to Skip)—the default is the total number of lines in the file minus the number of lines to skip, but if desired you can read in a smaller number of lines. (The lines to read can only be changed *after* hitting the Browse button). Then click on Read. If the file was correctly read in, you should be able to click on Apply, at which point your data will be plotted. To change the details of the plot, click the Advanced Features button which brings up the same Advanced Plot Features window which is described in



more detail in the Plot Panel section. Click on Cancel/Done to close the window. The shortcut key is (ctrl)-Shift-I (or Apple-Shift-I on Macintosh). In Batch mode see the IMPORT1D option.



4. **Save False Color Image:** After you have got the color scale, etc., set the way you want on the false color image, you can save the image in one of several standard graphics formats (PNG, JPEG, or TIFF). You will be prompted for the name (and location) of the file. A popup menu to the right or left gives you a choice of graphics formats. It is up to you to give your file name an appropriate extension (*.jpg, *.png); many programs, particularly in Windows, will not properly recognize a file with an incorrect extension. The jpeg format is suggested for most usages. However, some other applications may be happier reading in different format file. The program warns you if you are about to overwrite a previously existing file. The shortcut key is (ctrl)-S (or Apple-S on Macintosh). In Batch mode see the SAVEIMAGE option.
5. **Save 3D Image:** Saves the image created by the “3D Display” option in the plot image panel as a graphics file.
6. **Save 2D Data as ASCII File:** Most data files consist of binary data, possibly with an ASCII header or overflow buffer. This option allows you to save your data in ASCII format. This results in a file that can be more easily read by a user-written program (written, e.g., in C, Fortran, or a similar language) or by a text or spreadsheet program such as Word or Excel (if you have enough memory to read in the entire file). Keep in mind that the ASCII file will probably be about 10 times as large as the original data file, and it may take a while to write it out, depending on the size of the file and the speed of your machine. You can navigate through the file system to save the file where you want. You need to enter the name of the file in the box at the bottom—it is suggested that you choose .txt as an extension if you will be looking at the file on a Windows system. (On Linux this does not matter). On the right- or left-hand side there is a set of 3 buttons that

allow you to choose the delimiter between data columns: tab, comma, or space. In Batch mode see the SAVEDATA option.

- 7. Save 2D Data as Tiff File:** This option saves the raw data as a Tiff file. Since many data analysis programs, and even some manufacturer-supplied detector support programs, can read tiff files, this can be a useful option for those who want to export data saved in a manufacturer-defined format into a more commonly readable format. The data are saved in square, 16-bit tiff format. This is appropriate for most data formats, but does have some possibly unexpected consequences:

- If the data were originally saved in a floating point format, some information may be lost.
- Any negative data points (which can arise during post-processing by either the data-taking software or Datasqueeze) are replaced by zeroes.
- If the original bit depth is greater than 16 (i.e., if the largest data point is greater than 32767) then all data points are divided by 2^N such that the bit depth is reduced to 16.
- If the original data were in a compressed format, the final file size may be substantially larger.

In Batch mode see the SAVEDATAASTIFF option.

- 8. Save Plot Image:** Save the image of the x-y plot as a graphics file.
- 9. Save Plot Data as ASCII File:** If you have already created an x-y plot of the data, you can save the x-y data for subsequent manipulation by a plotting program such as Origin, Excel, Kaleidagraph, etc. You can navigate through the file system to save the file where you want. You need to enter the name of the file in the box at the bottom-- it is suggested that you choose .txt as an extension if you will be looking at the file on a Windows system. (On Linux this doesn't matter; on Macintosh it usually doesn't matter). On the right- or left -hand side there is a set of 3 buttons that allow you to choose the delimiter between data columns: tab, comma, or space. This matters to some plotting programs. The shortcut key is (ctrl)-Shift-S (or Apple-Shift-S on Macintosh). Note that the data are saved exactly as plotted--if you did a log-log or semi-log plot then it is the logarithm of the data point that is saved, not the raw value. The first three columns are always the independent variable, the plotted intensity, and the uncertainty. If you have done a least-squares fit to this data set then the function value and residual are plotted as well. In Batch mode see the SAVEPLOT option.
- 10. Export Plot Data:** Save plot data in one of several standard formats (cpi, csv, gsas, plv, dbws, ascii profile, ascii 2-column profile). In Batch mode see the EXPORTPLOT option.
- 11. Save Calculated Diffraction Lines:** If you have used the Show Powder Diffraction Lines feature in the plot window, you can save the calculated values of q and 2θ for each peak in a file.
- 12. Save Dataset Statistics as ASCII File:** If you have already calculated statistics for the current data set (using the Statistics window) you can save this information

in an ASCII file. Summary information is printed at the top, followed by tab-separated columns with bin statistics. As with the plot save option, it is suggested that you choose .txt as an extension. Note that the statistics are only refreshed when the Statistics window is opened (or, if it is open, if the Recalc or Autocalc button is clicked), so if you have changed files or binning options you will need to manually force a recalculation before saving the information to a file. In Batch mode see the SAVESTATISTICS option.

13. **Save Fitting Parameters as ASCII File:** If you have done a least-squares fit, you can save the values of the fitted parameters in an ASCII file. As with the plot save option, it is suggested that you choose .txt as an extension. Buttons allow you to choose between two types of file layout. If “Row” is selected, each fitting parameter will be on a different line in the file, with the error bars on the same line. If “Column” is selected, the fitting parameters are arranged by columns; the first line has the values, the next has the standard error, and the next two have the lower and upper limits (if calculated). There is also a checkbox for you to indicate whether or not you want to be able to append to an existing file; it is suggested for “safety” (to prevent damaging an existing file) you leave this unchecked unless you know for a fact that you want to append to a file. In Batch mode see the SAVEFITPARAMETERS option.
14. **Save Movie:** This option allows you to save a movie created with the Movie Controls option as a .avi file. You will be prompted for the file name.
15. **Page Setup:** Acts like a normal Page Setup menu item, allowing the user to choose paper size, orientation, etc. It is suggested that more attractive results will be achieved in Portrait orientation than in Landscape orientation. Use in conjunction with the Print command (below).
16. **Print:** The user has the choice of printing out a single summary page containing the false color image plus the line plot plus descriptive text, or the image only, or the plot only. It is obviously necessary to have an open data file for this to work and, if you want a plot, you should previously have generated the line plot from the Plot. You first get a dialog box which allows you to change the title that appears on the top of the page, and also modify the text that will be printed, which is automatically generated by the program. You can edit this text (Datasqueeze Software carries no liability if you lie about your data!) or add more descriptive text if you wish. A drop-down menu at the bottom allows you to choose between printing the entire summary page (the only case in which the text that you edited is actually used), just the image, or just the plot. When you are done, click the Done button. A normal Print Dialog window will then come up on Macintosh and Windows machines. The shortcut key is (ctrl)-P (or Apple-P on Macintosh). Results may vary on Linux or Unix machines--it appears that Java does not have very good support for printing on Linux, and whether printing works at all may depend on which flavor of Linux you are running (Red Hat, Debian, etc.). We hope to provide better Linux printing support in future versions. In Batch mode see the PRINT option.
17. **Save Current Settings:** Save information on your current settings for working folder, detector type, q-range, etc., for a future session. Note that this information is only saved if you click on this menu item. Note also that parameters such as q-

range are only retained in the next session if you set the “retain” button in the File panel.

18. **Restore Default Settings:** Should rarely be necessary. If you have gotten the system into some kind of unfixable state using the Save Current Settings menu item, this gets you back to where you started. Also, if you want to change the way you are formatting numbers (for example, from the American style to the German or French style) this will bring up a dialog box that will allow you to set your preferred style. After this, the program exits, and you will need to restart it.
19. **Exit:** Exit the program (not on Macintosh). The shortcut key is (ctrl)-Q (or Apple-Q on Macintosh).

Edit Menu

The Edit menu serves the normal functions of copying and pasting. For copying from or pasting to text boxes (i.e., numerical values), use the normal (cmd)C and (cmd)V commands (which cmd character to use depends on the operating system, but is the standard for that system). There are four special-purpose commands:

1. **Copy False Color Image:** Copy the false color image into the system clipboard, so that it can subsequently be pasted into another document. (PC and Linux only, deprecated on Mac OSX 9 and above).
2. **Copy Plot Image:** Copy an image of the plot panel into the system clipboard, so that it can subsequently be pasted into another document. (PC and Linux only, deprecated on Mac OSX 9 and above).
3. **Copy Fit Parameters:** Copy a text version of the fitting parameters into the system clipboard. They are tab delimited, so they are compatible with spreadsheet programs like Excel.
4. **Copy Fourier Transform Image:** If a Fourier Transform has been performed, copy the FT image into the system clipboard, so that it can subsequently be pasted into another document. . (PC and Linux only, deprecated on Mac OSX 9 and above).

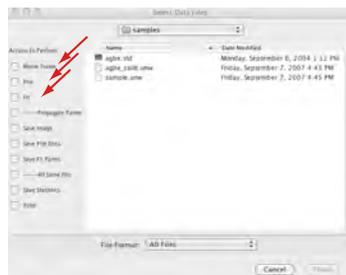
Batch Menu

Datasqueeze provides an interface for running in a “batch” mode, otherwise known as “scripting” or “running from a file”. The scripting file is a simple ASCII file that can be created with any text editor. It should be raw text--that is, it should not contain any formatting commands (bold, etc.) such as those produced by Word or most other text editors. It is suggested that the file have a suffix .txt, since this will indicate to Windows and Macintosh systems that this is a plain text file, but in fact Datasqueeze itself doesn't care. This feature is primarily useful if you want to process a number of data files in the same way, without visually examining each one. For example, you might want to read in a set of files, and for each one save a false-color image and an ASCII version of the plot.

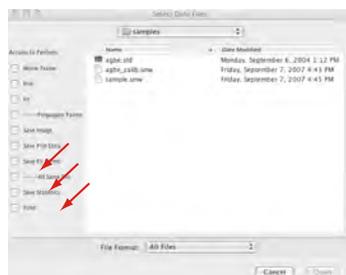
There are presently four options under the Batch menu:

1. **Enable Logging Commands:** This enables output to a text file of the commands you have entered from the graphical interface. A window opens which prompts you for the name of the file. It is suggested that you give it a .txt suffix. The shortcut key is (ctrl)-L (or Apple-L on Macintosh).
2. **Disable Logging Commands:** This turns off command logging if it has been previously turned on. The shortcut key is (ctrl)-Shift-L (or Apple-Shift-L on Macintosh).
3. **Read from Command File:** This causes the program to take its input from a command file that you specify. Note that the syntax of these command files is identical to that of the log files that you create with Enable Logging Commands, so a convenient way to create a command file is to enable logging, perform a set of typical commands, and then use any standard text editor to modify the log file and create the commands that you want. The shortcut key is (ctrl)-B (or Apple-B on Macintosh).
4. **Process Multiple Files** This is an alternate approach to file processing, in which multiple files (anywhere from one to thousands) are treated in exactly the same way. When this menu item is selected, a dialog box is opened which allows you to select one or multiple files to process, and to check off various options. For most of these, default parameters are chosen based on your most recent actions. If data are saved, a default file name is chosen based on the name of the data file. In the following description, we assume that your data file name is FILENAME. The possible options are:

- **Movie Frame:** If selected, will add the false color image to the buffer of frames to show in a movie.
- **Plot:** If selected, will make a linear plot of the last type done (e.g., Q/Chi) with the ranges and options previously selected. You need to make a plot if you plan to do a least-squares fit to the data.
- **Fit:** If selected, will do a least-squares fit to the data using a previously initialized fit model, starting parameters, and selection of which parameters to fit.



- **Propagate Parm:** This affects how the parameters are chosen for the least-squares fit. If checked, each fit will use the as its starting point the fitted parameters from the last data set. This is useful, for example, if you are analyzing a temperature series where there is some kind of slow variation (e.g., a peak moving) but each data set resembles the previous one. However, if a fit fails for some reason then all subsequent fits will probably start with nonsense parameters. If this box is not set, then each



data set is analyzed with the same starting parameters--"safer," but not as good if there is some kind of slow variation in peak position, width, or other parameters.

- **Save Image:** If selected, the false color image will be saved as a graphics file in whichever format (JPEG, PNG, TIFF) you last used. The filename will be FILENAME_img.jpg, FILENAME_img.png or FILENAME_img.tif. The image resolution will be set by the "Saved Image Resolution" radio button in the Image panel.
- **Save Plot Data:** If selected, and if you have done a linear plot, the plot data will be saved as a text file called FILENAME_plot.txt.
- **Save Plot Image:** If selected, and if you have done a linear plot, an image of your plot will be saved as a graphics file, in whichever format (JPEG, PNG, TIFF) you last used, called FILENAME_plot_img.xxx, where xxx is one of jpg, png, or jpeg.
- **Save Fit Params:** If selected, and if you have done a least-squares fit, fit parameters will be saved as a text file called FILENAME_fitp.txt.
- **All Same File:** If you are saving fit parameters, and this box is checked, all the parameters will be saved in the same, initial file. Otherwise, the parameters for each data file will be saved in a different file.
- **Save Statistics:** If selected, data set statistics will be saved in a file called FILENAME_stat.txt.
- **Print:** If selected, a summary fit page will be printed. Note that on most operating systems this brings up a dialog page whether you want to or not, so some user intervention is required.

As most of these commands take their parameters from previous operations, it is important that you have previously opened a data file similar to the ones you will be reading in, done a plot, and done a least-squares fit with final parameters similar to those you will use. You will probably want to set the "Retain" option in the File panel. A good way to accomplish all of this is to have a batch file set up that processes just one file in the way you want to do the rest. Any saved files will appear in the same directory (folder) as the original data files. If a file with the same name already exists, the procedure comes to a halt. So, if you run Process Multiple Files twice in the same directory, you may need to remove some files in between. Also, the procedure clearly will not work if the files sit in a write-only directory (for example, on a CD). One good way to set up for processing files would be to copy all of the data files you want to file into a new folder. Then you could select all of the files in that folder to be processed, use the check boxes to select what you want to do, and then process all the files--in principle a very large number.

View Menu

There are presently six options under this menu:

1. **File Info:** This opens a screen providing possibly useful information about the current data file. Note that the information provided does not reflect any changes made in the image center, counts added or subtracted when read in, etc.--it is the “as-read” information from the first file in the list. There is currently no way of displaying information about multiple files that have been added--you would have to open each file individually. The shortcut key is (ctrl)-I (or Apple-I on Macintosh).
2. **Format Info:** This opens a screen providing information on the available file formats--what hardware probably created it, what is inside, and what kind of processing Datasqueeze does.
3. **Statistics:** This opens a screen with information on pixel statistics. The shortcut key is (ctrl)-T (or Apple-T on Macintosh). In the default configuration, the program calculates the number of pixels with total counts in each of 20 bins, ranging from the least intense to the most intense. Both the total number of pixels, and the percentage of pixels, in each bin are calculated. This is done both for the entire data set, and for the region selected in the Plot. This allows the user, for example, to exclude strong scattering near the beam stop from the statistics. If desired, new bin ranges can be entered by editing the low and high ranges for each bin. The bins do not have to be in order, and they can even overlap. To recalculate the statistics using your new, edited ranges, click on the Recalculate button. To restore the default binning ranges, click on the Autocalc button. Boxes at the bottom show the minimum, maximum, and average counts per pixel in the entire data set, and for the region selected in the Plot Panel. These are straightforward. There is also a box labeled “Dispersion”. This is more complicated. It calculates



$$D = \text{avg} \left(\frac{(y_i - E_i)^2}{E_i} \right)$$

where y_i is the actual counts in pixel i , E_i is the average of the counts in the four surrounding pixels (i.e., a measure of the “expected” counts), and the average is done over pixels with 50 or more counts. This parameter has various interpretations: Suppose first that the data are fairly flat and featureless (for example, produced by a “flood field” calibration source) and that the counts in each pixel are actually equal to the number of photons that hit that part of the detector. In that case, E_i should actually be the expected number of counts in the pixel in question, and if Poisson statistics hold then D should be equal to 1.

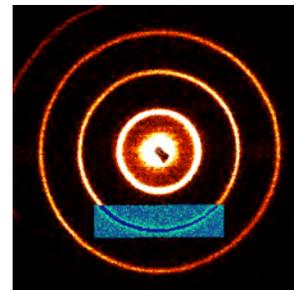
Suppose instead that the signal is still featureless, as above, but the counts reported for each pixel are only proportional to the number of photons that hit that pixel, $y_i = a N_i$. In that case you can show that the parameter D is actually a measurement of the proportionality constant “ a ”. On the other hand, if there is curvature to the data (for example, if the data set contains Bragg peaks, not unknown for x-ray data sets!) then D will be greater than 1 even if Poisson statistics hold. In this case the significance of D is somewhat less clear--it is a measure of the curvature, or second derivative, of the data set. In Batch mode, see the `CALCSTATISTICS`, `NUMSTATBINS`, `SETSTATBIN`, and `SAVESTATISTICS` commands.

4. **Show Access Key:** This menu item opens a small window showing your access key to Datasqueeze. This is useful if you have forgotten to write it down and want to enable Datasqueeze on a different computer.
5. **Zoom Out:** Demagnify the false color image by a factor of 2. Also accomplished (with more options on centering) via the Zoom button in the Image panel. See `ZOOM` command in batch mode. The shortcut key is (ctrl)- (or Apple- on Macintosh).
6. **Zoom In:** Magnify the false color image by a factor of 2. Also accomplished (with more options on centering) via the Zoom button in the Image panel. The shortcut key is (ctrl)+ (or Apple= on Macintosh). See `ZOOM` command in batch mode.

Mask Menu

A “mask” allows the user to exclude certain pixels from plots and calculations. These might be known “bad” pixels, scattering near the beamstop, scattering near known spurious features, etc. By default, pixels outside the data collection area are excluded, but these boundaries can be moved in. The user has the option of adding additional masks to exclude data inside specified rows or columns, individual pixels, inside a user-defined box or circle, or outside a user-defined circle. There are presently four items under this menu:

1. **Show/Hide Masks:** If “Show Masks” is checked, the excluded areas are shown in a different false color palette (which is more or less the complement of the original palette). If the masked regions are shown, then checking “Hide Masks” will remove the highlighting.
2. **Edit Masks:** Checking this item brings up a new window with controls for editing current masks or creating new masks.
 - a. To edit a current mask, either type the new desired value(s) in the box, or drag



the blue cursor dot corresponding to that mask. For example, the first mask always excludes points outside the left-hand side of the image, and the default setting is to exclude pixels with $x \leq -1$. To create a new mask, select the desired mask type from the drop-down menu and then click Apply. Currently available options are: “Exclude Points In Row”, “Exclude Points In Column”, “Exclude Points In One Pixel”, “Exclude Points Inside Box”, “Exclude Points Inside Circle”, “Exclude Points Outside Circle”.

- b. To use the new masks, click Done.
 - c. To revert to previously used masks, click Cancel.
3. **Save Masks:** A set of defined masks can be saved in a file for subsequent use. This is most useful if the same masks will be used over and over for many data files. Selecting Save Masks brings up a dialog that allows the user to select the name of the file where the mask information will be saved. We suggest that the file be given a suffix of .txt or something similar since it is a text file. The format of the created files is quite straightforward, and the user could also create them himself/herself using a text editor. If this is done, keep in mind that the “pixel” mask is relatively inefficient. Any time a data-intensive operation such as making a plot is performed, Datasqueeze must query each mask, so having a large number of individual masks can result in a substantial degradation of performance.
 4. **Import Masks:** This option retrieves a previously saved mask.

To use these features in Batch mode see the MASK, APPLYMASK, and NMASK options. Note that the user always starts off with four masks that should not be changed, which tell Datasqueeze not to use data outside the active area of the detector. So, for example, for a 1024 x 1024 file, the first 4 masks would be specified as

```
>MASK MINXMASK 0 -1.0000000  
>MASK MINYMASK 1 -1.0000000  
>MASK MAXXMASK 2 1.0240000E3  
>MASK MAXYMASK 3 1.0240000E3
```

This indicates that no pixels should be considered with x or y less than 0 or greater than 1023. Other user-defined masks would then start with mask number 4, e.g. something like

```
>MASK PIXELMASK 4 247.00 825.00
```

Tools Menu

There is presently only one item under this menu:

1. **Q - S - D - Theta Calculator:** This convenient tool allows the user to quickly convert between different units for describing diffraction data.

a. Wavelength: this is the wavelength (λ) of the incident radiation. Although it is initialized to the wavelength associated with the current data set, it can be set to any physically meaningful value desired. Changing the wavelength in this window will **not** change it elsewhere in the program.

$Q = (2 \text{ Pi } S) = (2 \text{ Pi } / D) = (4 \text{ Pi } / \lambda) \sin(\theta)$

Wavelength	1.54056
Q	0.77999
S	0.12414
D	8.05542
Theta	5.48715
2-Theta	10.97431

b. Q: This is the momentum transfer, in the units most often used by physicists:

$$Q = \frac{2\pi}{D} = 2\pi S = \frac{4\pi}{\lambda} \sin\theta$$

c. This is also the momentum transfer, in the units often used by crystallographers:

$$S = \frac{Q}{2\pi} = \frac{1}{D} = \frac{2}{\lambda} \sin\theta$$

d. D: This is the d-spacing associated with a particular momentum transfer (not necessarily a actual plane spacing in any crystal):

$$D = \frac{1}{S} = \frac{2\pi}{Q}$$

e. Theta: this is the angle between the incident beam and the scattering plane, in degrees.

f. 2-Theta: this is the angle between the incident and outgoing rays, in degrees.

Help Menu

There are presently seven items under this menu:

1. **General Information:** This menu item opens a window showing the help file, similar to the manual section you are presently reading. The shortcut key is (ctrl)-H (or Apple-H on Macintosh).
2. **Batch Help:** This opens a separate help window with detailed instructions on the command file syntax, as well as a brief description on how to use batch command files, with a simple example.
3. **Fit Help:** Provides more detailed information and advice on least-squares fitting, including a sample session. This information is also reproduced later in this manual.
4. **Sample Session:** A tutorial introduction to the program, with a sample session.
5. **FAQ:** Answers to frequently asked questions.

6. **Version History:** Provides a summary of changes made to each version of Datasqueeze
7. **Check for Updates:** Checks online to see if a newer version of Datasqueeze is available

Windows Menu

This menu provides a list of current open Datasqueeze windows. You can use it to navigate to a different window (for example, going from the Fit window to the false color image). You can also move from one window to another by using ctrl-~ (Apple-~ on Macintosh).

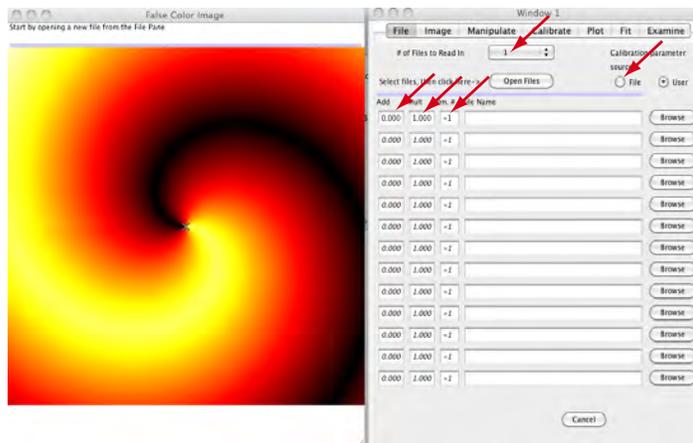
Sample Session

This section leads you through some of the actions you might take in a typical data analysis section.

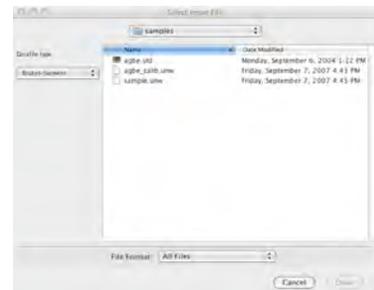
Setting Calibration Parameters

We begin by opening a calibrating the detector parameters.

- Double-click the Datasqueeze icon to start the program.
- In the File panel verify that the number of files is set to 1 and “Calibration Parameter Source” is set to “File.” Verify also that for the top set of boxes “Add” is set to 0, “Mult” is set to 1, and “Im. #” is set to -1.

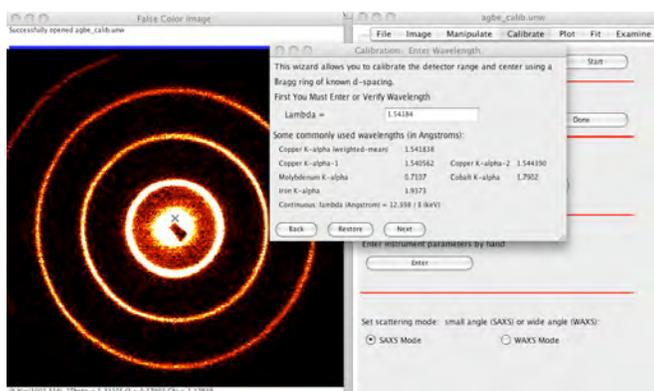


- Use the Browse button to go to the “samples” folder that was created when you installed the program. This will probably be in C:\Program Files\samples\samples on Windows, /usr/src/datasqueeze_files/samples on Linux, or Macintosh HD/Applications/datasqueeze_files/samples on a Macintosh.
- Set the data type to “Bruker-Siemens”.
- Select the file called agbe.std by selecting it and clicking on Open. The open file dialog box should close.





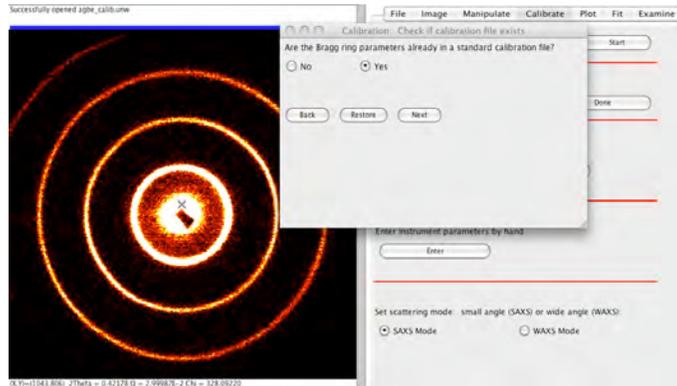
- A new window opens which asks you to set the correct wavelength. For the file in question this should be Cu K-alpha wavelength, 1.54184. If it is not, type the correct number in the box. Then click Next.



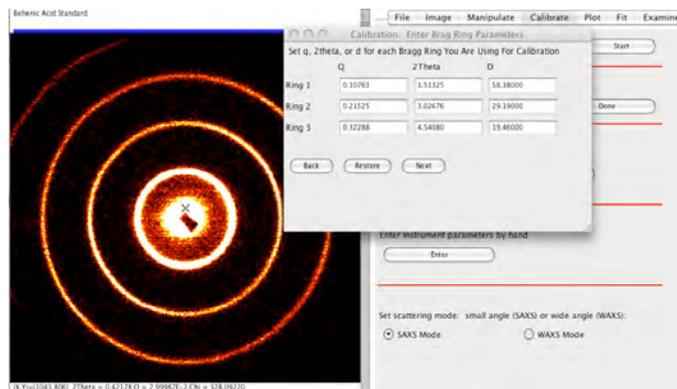
- The next window asks us to set the pixel size or detector dimensions. Since we are not sure of these values for our detector, we follow instructions and leave them as they are. Click Next.
- Now we set the number of Bragg rings, which is three for this particular file. Set the drop-down box to 3 and click Next.



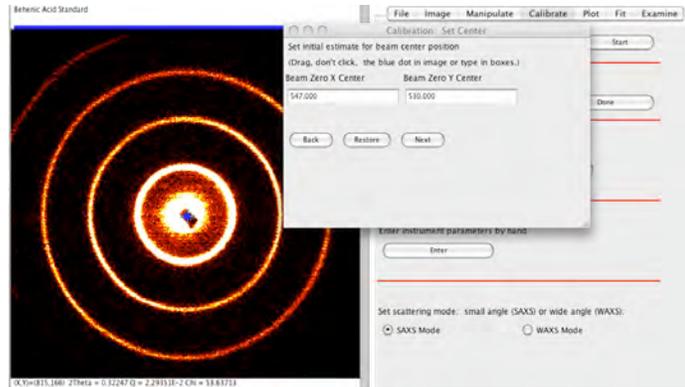
- The parameters are indeed in a standard calibration file. Set the “Yes” check box and click Next.



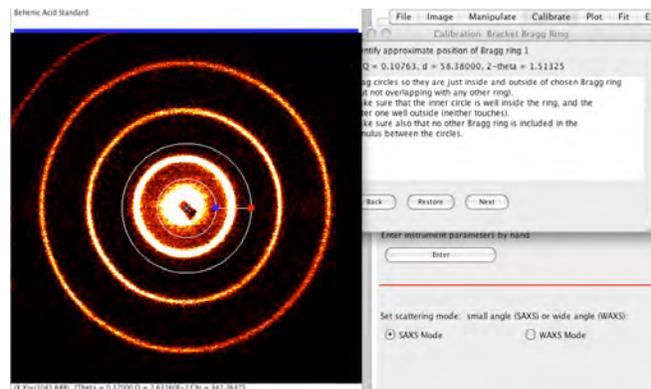
- In the dialog box that opens, select agbe.std. This sample file is silver behenate, a lamellar material with a d-spacing of 58.3 angstroms.
- The next window displays the Q, 2-theta value, and d-spacing for the rings we chose. If we had not read in the agbe.std file, we could have entered any of these by hand. Click Next.



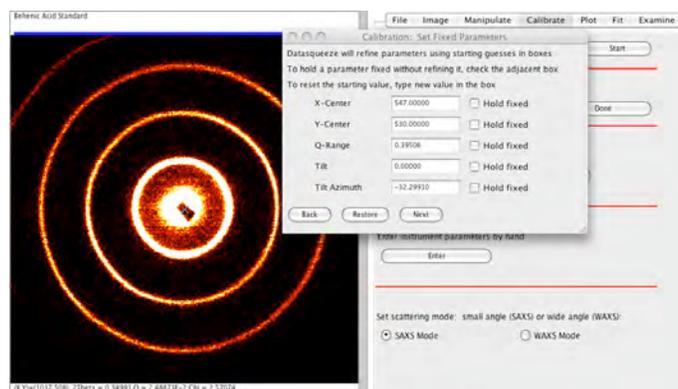
- We are now asked to set the estimated position of the beam center. Drag the blue dot in the false color image until it is approximately in the center of black dot in the middle of the image. In the boxes, X and Y should be approximately 547 and 530. Then click Next.



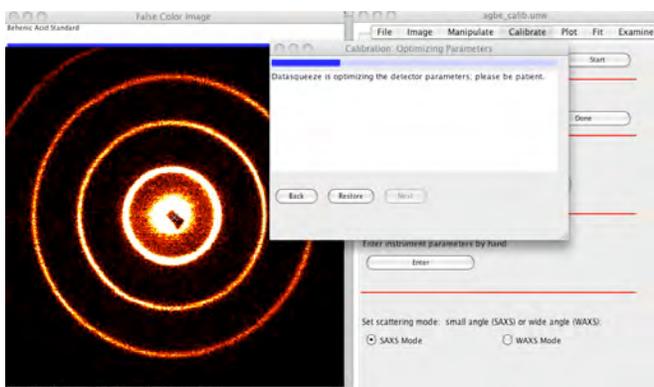
We now need to tell Datasqueeze where the first ring is that we will use for calibration. Drag the outer blue dot so it is just outside the innermost data circle. Drag the inner blue dot so it is just inside that circle, and click Next. Repeat for the second and third rings.



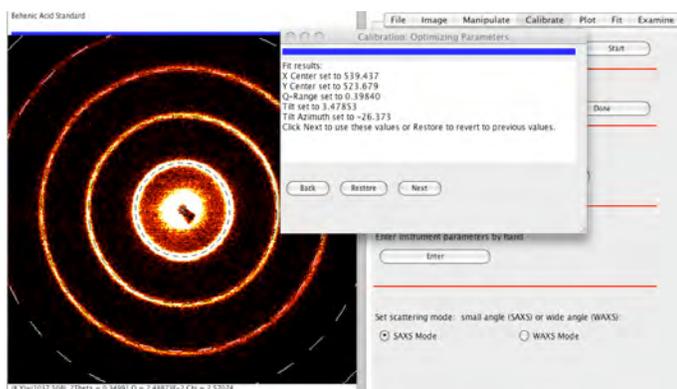
- We now get a window asking us to verify the starting parameters. Click Next.



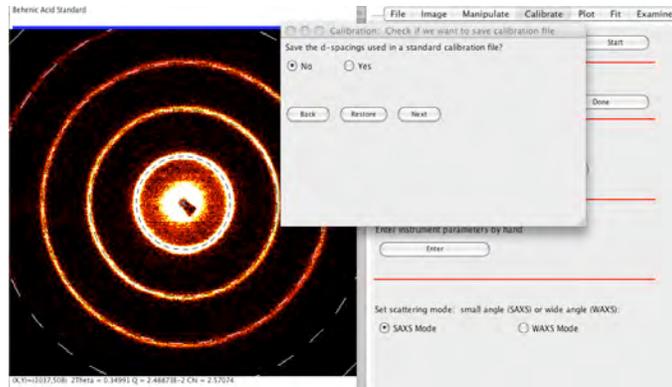
- Datasqueeze will now do a series of least squares fits to determine the optimum detector parameters. This may take several minutes depending on the speed of your computer.



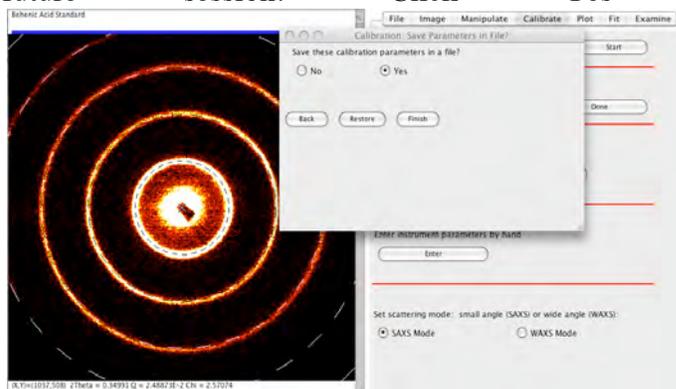
- When the calibration is finished, the results should appear in the white box. You should get something like: X Center set to 539.462, Y Center set to 523.801, Q-Range set to 0.39784, Tilt set to 4.56488 and Tilt Azimuth set to -32.299. The X Center and Y Center define the center of the diffraction pattern (not necessarily exactly where your beam stop is). In this file they are close to the geometric center of the image. The Q-range is a measure of the angular scale of your instrument. The Tilt reflects the fact that, either due to electronic anisotropies or detector misorientation, your patterns might not be exactly circular. On the false color image, in addition to the data, you will see black-and-white dashed circles; verify that one of those circles is in fact directly on top of the Bragg ring.



- If you are happy with these results (we are) click Next. You could also click Restore to revert to former calibration values or Back to back up one or more steps.
- We are now asked if we want to save these d-spacings. Since we already read the data in from a calibration file, the answer is probably “no” but if we had typed them in by hand this would be our chance to create our own calibration file. Click Next.



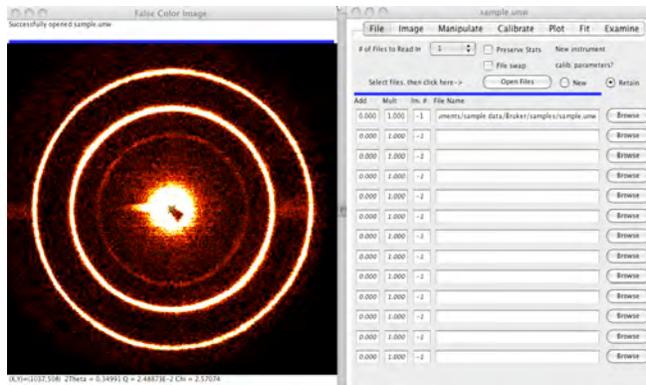
- We are now asked if we wish to save these calibration parameters in a file. If we check Yes then we will create a file with information on the current session, which we could use to avoid repeating the calibration process in a future session. Click Yes and Finish.



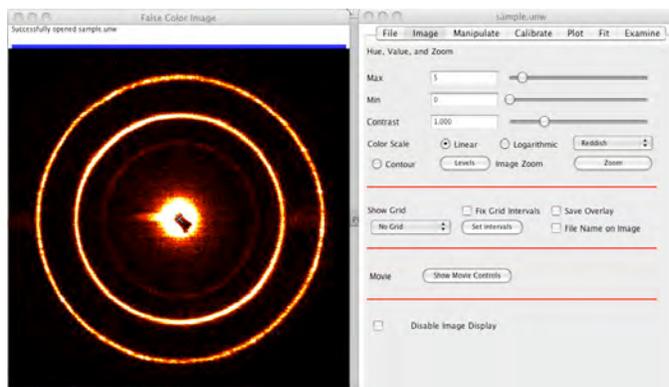
Look at some real data

Now let's look at a real data file.

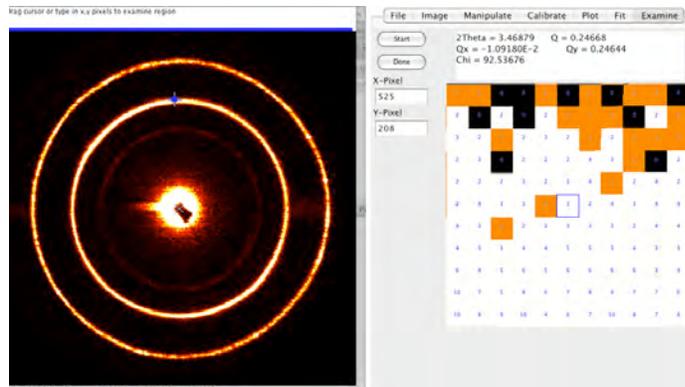
- Go back to the In the File panel. As before, select (with the Browse button) and open (with the Open Files button) the file called sample.unw. You should observe multiple rings.



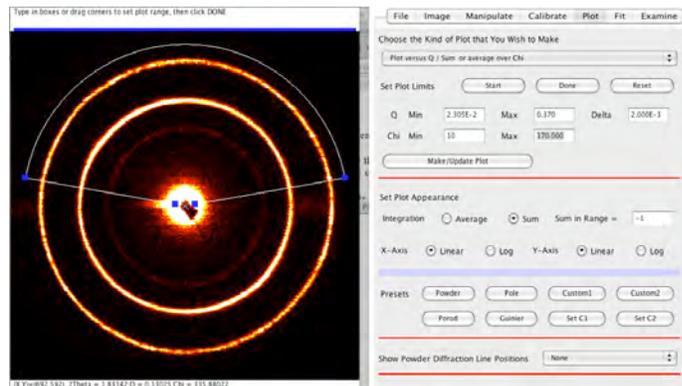
- In the Image Panel, highlight the text in the Maximum for False Color box, set it to 5, and hit a carriage return. The inner rings become more distinct while the outer rings become darker.



- Click on Logarithmic to see what that does, then go back to Linear.
- This is a nice image; we might want to publish it. Go to the FILE Menu and select Save False Color Image.
- The file chooser dialog box opens. Use the directory scroll button near the top to choose a better folder, maybe your home directory (on Linux) or the desktop (on Mac or Windows) so that you can find this thing to delete it later.
- Let's save it as a JPEG file--verify that the drag- down menu to the left is set correctly.
- In the Name box type tmp.jpg. Click on the Save button.
- Let's examine a region of the plot more closely. Go to the Examine Panel and click the Start button.
- An X with a blue center appears near the center of the main image.
- Click on the X and drag it to see a pixel-by-pixel blow-up of different regions of the image.

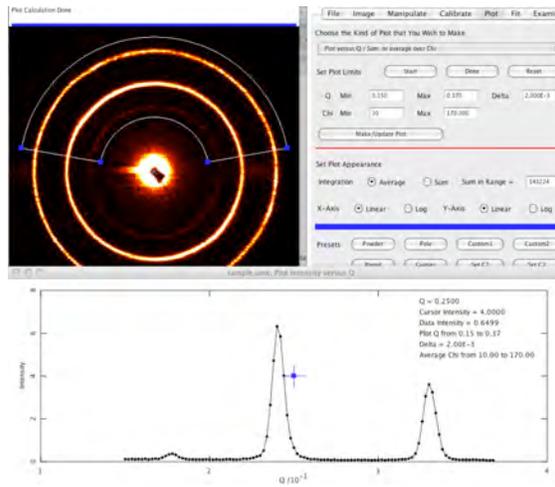


- Now Click Done.
- Go to the Plot. Let's do a Q plot that emphasizes the top part of the pattern. With the drop-down menu, select *Plot versus Q / Sum or average over Chi* (it is the default, so it is quite likely already selected).
- Click on the Start button for Set Plot. Either by dragging the blue dots, or by entering directly in the relevant boxes, set up a region that covers all but the innermost ring. We found that $Q_{min}=0.15$, $Q_{max}=0.37$, $Chi_{min}=10$, $Chi_{max}=170$ worked well.

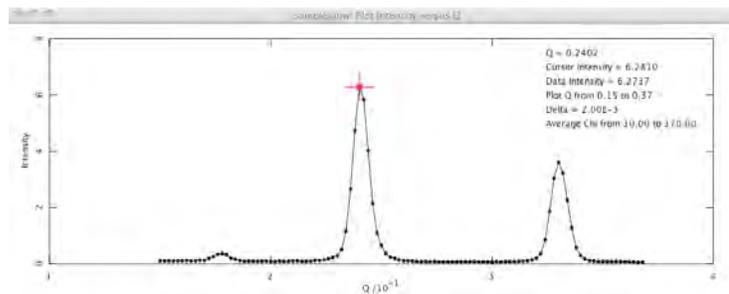


- We also need to set Delta. Let's try $Q\text{-Delta} = 0.002$
- Set the Integrate button to Average.

- Let's see what this looks like. Click Make/Update Plot. You should see one weak peak and 2 strong peaks in the Line Plot Image window.



- Select the plot cursor (it should turn from blue to red) and drag it over to the top of the strongest peak. The readout in the upper right-hand corner of the Line tells us that this peak is located at $Q=0.2402$ and has an intensity of 6.2.



- Set the Integrate button to Sum and click again on Make Plot. Notice that the higher-Q peak got stronger, relatively speaking.
- Let's save these data. Go to the FILE Menu and select Save Plot Data as ASCII File. Choose the Comma button for Set Separator. Enter the name as "test.txt".

- Temporarily leave the program and use your favorite text editor to look at the file you got: three columns of ASCII numerical data.

```

0.1500 164.0000 12.8062
0.1520 152.0000 12.3288
0.1540 173.0000 13.1529
0.1560 155.0000 12.4499
0.1580 176.0000 13.2665
0.1600 160.0000 12.6491
0.1620 198.0000 14.0712
0.1640 158.0000 12.5698
0.1660 182.0000 13.4907
0.1680 213.0000 14.5945
0.1700 200.0000 14.1421
0.1720 309.0000 17.5784
0.1740 420.0000 20.4939
0.1760 551.0000 23.4734
0.1780 597.0000 24.4336
0.1800 535.0000 23.1301
0.1820 346.0000 18.6011
0.1840 273.0000 16.5227
0.1860 214.0000 14.6287
0.1880 177.0000 13.3041
0.1900 186.0000 13.6382
0.1920 183.0000 13.5277
0.1940 181.0000 13.4536
0.1960 188.0000 13.7113
0.1980 194.0000 13.9284

```

- Let's find out some statistics on this dataset. Select Statistics in the View menu. Average, maximum, and minimum counts are calculated for the dataset as a whole and also for the region selected for plotting, and the number of counts in each of 20 different bins are calculated. If we wanted to know more we could reset the number of bins and the bin ranges and recalculate all of this in more detail.



- Let's make a souvenir printout. Select Print in the File menu.
- Just for fun, change the X and Y Axes from Linear to Log and click the Make Plot. The plot should look quite different--the weak peaks are emphasized much more in this plot.

This concludes the tour of the program. You have used most of the features, and the ones that you have not (such as Zoom) are pretty obvious. Have fun analyzing data!

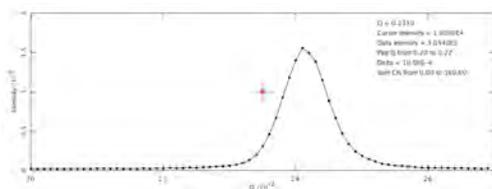
Help With Least Squares Fits

This section of the manual is divided into four parts. **Getting Started** provides a quick guide to performing a least squares fit using *Datasqueeze*. It will be useful for new users who want immediate gratification, or experienced users who have not used this feature for a while and need a quick reminder. **Advice on Least Squares Fits** provides a longer description of least squares fitting, including numerous warnings about possible pitfalls. You should definitely read this section before showing the results of any least squares fits to your boss or advisor! **Technical Details** provides a bit more information about what goes on “under the hood.” It will be primarily of interest to those who have previously used (or even written) least-squares fitting algorithms and want to understand in more detail how the program works. Finally, **Functions Provided** gives some information on the functions currently available in *Datasqueeze* for least-squares fits.

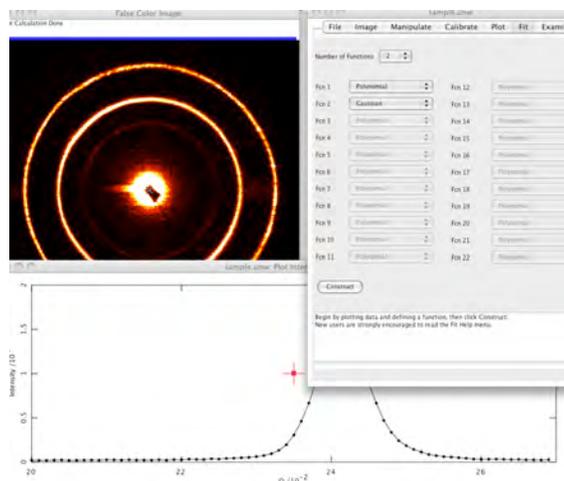
Getting Started with Least-Squares Fits

In this section it is assumed that you are already familiar with the important windows and controls in *Datasqueeze*. The general procedure for performing a least-squares fit in *Datasqueeze* is as follows:

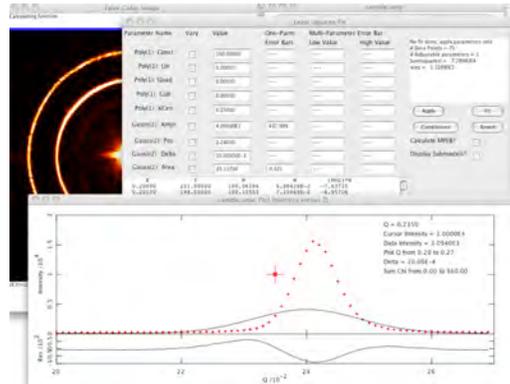
1. Open a data file in the usual way, either with (cmd)O or by going to the File panel.
2. Make a line plot of your data in using the Plot panel. You can plot versus any parameter you want--for example, plot intensity versus Q or versus Chi. However, for some models (e.g. a Rayleigh lineshape), only plotting versus Q makes much sense.



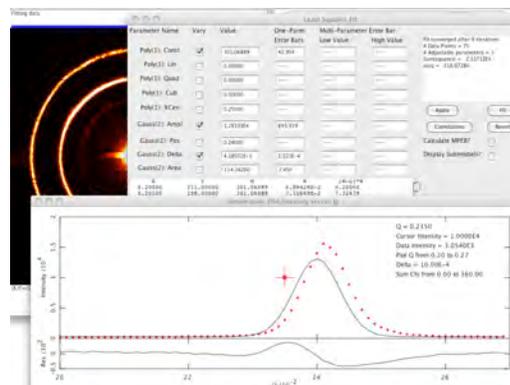
3. The fit panel allows you to construct a model function as a sum of provided subfunctions. You specify the number of subfunctions, and what you will use for each. For example, if you wanted a Gaussian peak with a linear background, you would select 2 for the Number of Functions, Polynomial for the first subfunction and Gaussian for the second subfunction.



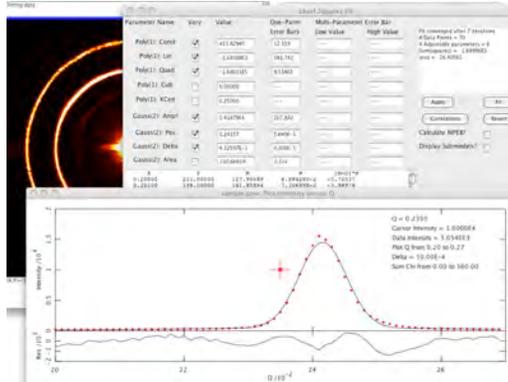
4. Some functions (e.g. the Rayleigh function for small-angle scattering from spheres) expect Q as the independent variable. You can use these models even if you have plotted data versus 2θ or Q^2 by checking the “Use Q as Fit Variable” box. This will be true for all submodels used, even those such as Gaussian which do not necessarily require Q as the independent parameter. Do not check this box if you want to be sure that the variable you used for plots to be that used for the fit itself.
5. Once you are happy with the way you have set things up, click on the Construct button to bring up the Fit Parameter panel.
6. If you have not previously used a model, Datasqueeze sets the starting parameters to reasonable values, consistent with the way the data have been plotted. For example, the default parameters for a Lorentzian peak will place the peak somewhere in the region of your plot, with an amplitude comparable to the scale of your data and a width that is a small fraction of the total width of your plot. However, the starting peak position, amplitude, and width will still probably be nowhere near those of the actual features in your data. Therefore, you need to set each parameter to a reasonable starting value. This is important--the fitting algorithm (as with any least-squares algorithm) will only work if your starting parameters are reasonably close to the “true” values. Setting initial values can be done in two ways:
 - a) Most fit models come equipped with cursors, which allow you to graphically change some or all of the fit parameters. You can drag the cursors around until the calculated shape resembles that of your data. If you do not see the cursors the first time, you may need to click the Reset button.
 - b) For parameters that do not have corresponding cursors, or for finer control, type into the appropriate box in the Fit Parameter panel to set each parameter to a reasonable starting value. After doing this, click the Apply button. Your function will be plotted on top of the data in the Line Plot Image window. See whether your model with the starting parameters you have chosen agrees approximately with the data.



- It is better not to let all the parameters vary simultaneously, unless you have a particularly simple model. Rather, you should pick two or three parameters, vary them, and then gradually add more. So, at this point you should check off a few parameters that you want to vary.
- Click on the Fit button to vary the checked parameters.



- If you have chosen wisely, the agreement between model and data has improved, and the Message area in the Fit Panel does not show an errors. If so, go back to step 8, check a few more parameters, and continue until all the parameters that you wish to vary have been optimized. (Note that you do not have to minimize every parameter--if you have other knowledge about some parameters you may prefer to hold them fixed).



10. If you get unexpected or unwanted results, you may want to deselect a few parameters, click on the Revert button, and try again.
11. If you keep getting error messages, you may wish to click the Correlations button and see whether two or more parameters are strongly correlated.

Parameter Correlation Coefficients

	Poly(1): Const	Poly(1): Lin	Poly(1): Quad	Gauss(2): A...	Gauss(2): Pos	Gauss(2): D...
Poly(1): Const	1.000	-0.688	0.798	0.126	-1.042E-2	0.214
Poly(1): Lin	-0.688	1.000	-0.938	-3.406E-2	4.045E-2	-6.202E-2
Poly(1): Quad	0.798	-0.938	1.000	1.064E-2	-1.717E-2	2.595E-2
Gauss(2): A...	0.126	-3.406E-2	1.064E-2	1.000	4.815E-3	0.616
Gauss(2): Pos	-1.042E-2	4.045E-2	-1.717E-2	4.815E-3	1.000	-2.804E-2
Gauss(2): De...	0.214	-6.202E-2	2.595E-2	0.616	-2.804E-2	1.000

12. A correlation coefficient with magnitude greater than +/- 0.9 is not good, and a correlation coefficient with magnitude greater than +/- 0.98 almost certainly indicates a real problem. In this case you may need to either use different starting parameters or hold more parameters fixed.
13. The one-parameter error bars give an indication of the uncertainty in each parameter.
14. A better estimate is provided by checking the “Calculate MPEB?” and redoing the fit. You may however find that some parameters are strongly enough correlated that multi-parameter error bars cannot be calculated, in which case you need to deselect those parameters and redo the fit without varying them.

Parameter Name	Vary	Value	One-Param Error Bars	Multi-Parameter Error Bar Low Value	Multi-Parameter Error Bar High Value
Poly(1): Const	<input checked="" type="checkbox"/>	413.52963	12.359	390.34460	437.31126
Poly(1): Lin	<input checked="" type="checkbox"/>	-2.63084E1	163.782	-3.80199E1	-1.48376E1
Poly(1): Quad	<input checked="" type="checkbox"/>	-1.69031E1	5.514E1	-2.04899E1	-1.30924E1
Poly(1): Cub	<input type="checkbox"/>	0.00000	---	---	---
Poly(1): XLen	<input type="checkbox"/>	0.75000	---	---	---
Gauss(2): Ampl	<input checked="" type="checkbox"/>	1.41879E4	207.682	1.39073E4	1.44694E4
Gauss(2): Pos	<input checked="" type="checkbox"/>	0.24157	5.649E-1	0.24151	0.24162
Gauss(2): Delta	<input checked="" type="checkbox"/>	4.32197E-3	4.300E-3	4.26147E-3	4.33047E-3
Gauss(2): Area	<input type="checkbox"/>	230.66658	2.324	228.64223	232.69093

X	Y	H	W	(H-D)*W
0.20000	211.00000	127.49677	6.88428E-2	-8.72037
0.20100	198.00000	141.85813	7.10669E-2	-3.86979
0.20200	193.00000	155.47463	7.19183E-2	-2.70114
0.20300	221.00000	168.75447	6.72672E-2	-3.51441
0.20400	239.00000	181.69822	6.40846E-2	-3.70824
0.20500	190.00000	194.10597	7.22475E-2	-6.31239
0.20600	204.00000	206.57763	7.00110E-2	-0.28047
0.20700	230.00000	218.51322	6.59380E-2	-6.75742
0.20800	230.00000	230.11276	6.59380E-2	-7.41500E-1
0.20900	215.00000	241.37623	6.81994E-2	1.79884
0.21000	253.00000	253.39722	6.32000E-2	-7.19376

15. If you are happy with the results, you may want to print the plot showing the agreement between model and data (starting from the File menu), or save the fitting parameters in an ASCII file (again starting from the File menu).

Advice on Least Squares Fits

How Least-Squares Fitting Works

- Suppose you have measured some quantity y as a function of some independent parameter x . You have done this at N different points, so you have N pairs of values (x_i, y_i) , each with an uncertainty in y_i given by e_i . You believe that you can describe this with a model $f(x)$ which contains M independent parameters b_j . You want to find the values of b_j which provide the best agreement between the model and the data.
- The “goodness of fit” parameter that you use to describe the agreement is called ϕ :
- The goal is therefore to find the set of parameters b_j which minimize ϕ . There are different algorithms for accomplishing this, but they all rely in some way on taking numerical or analytical derivatives of ϕ with respect to each of the b_j , and then iteratively adjusting the values of each of the b_j until the minimum in ϕ is found.
- The problem can be compared to the case of a lost hiker. The hiker tries to find his way back to civilization by always heading downhill, i.e., heading towards his best guess for the “minimum in altitude.” This is problem in minimizing one parameter (the altitude) in two dimensions (the two-dimensional surface of the Earth). The difference in least-squares fits is that the parameter space in general contains more than two parameters.
- The least-squares algorithm used by *Datasqueeze*, like most others, has the following properties and limitations:
 - It finds a local minimum in ϕ , which is not necessarily the global minimum. Thus, if inappropriate starting parameters are chosen, the final fitted values of the parameters may not provide a good fit to the data.
 - Since derivatives and other functions are calculated numerically, there is always some numerical uncertainty in the final values. However, this uncertainty is normally much less than the statistical uncertainties in these parameters.
 - Since the program cycles through a finite number of iterations, it is possible that it may never find the local minimum, particularly if the function is quite insensitive to some of the parameters.

What Chi-Squared Means

- The goodness-of-fit parameter obviously depends on the number of data points. A better parameter to describe the agreement between model and data is χ^2 . Definitions of χ^2 vary, but the definition used here is:

$$\chi^2 = \frac{\Phi}{N - M}$$

- where, again, N is the number of data points and M is the number of parameters that were varied. (This may be different from the total number of available parameters if some of them were not allowed to vary.)
- Remembering that each data point has a statistical uncertainty e_i , we can see that if the model describes the data “perfectly” an average calculated value $f(x_i)$ will differ from the “true” value by e_i , and χ^2 will be close to 1.
- If $\chi^2 \gg 1$, it can mean one of several things:
 - Although your model may visually resemble the data in a plot, there are still statistically significant differences between the two.
 - You have underestimated your error bars. Datasqueeze assumes that the error bars are all given by Poisson statistics, that is, that the uncertainty in a data point is approximately the square root of the number of counts. This is a fairly good assumption for photon-counting detectors, but may be a terrible assumption for other types, such as image plates.
- If $\chi^2 \ll 1$, it also can mean one of several things:
 - You have too many fitting parameters, so that your model is actually “tracking the noise.”
 - You have overestimated your error bars (see above).
- In general, you hope that χ^2 decreases as you minimize more and more parameters. If it stays the same or increases, it means that any improvement in ϕ is statistically insignificant. And, obviously, if the number of data points is less than or equal to the number of fitting parameters then the fit has no meaning.

What the Error Bars Mean

Since there is uncertainty in the data, there are obviously uncertainties in the fitted parameters. There will always be a spread in parameters which will give a result that is statistically indistinguishable from the “best” result. Datasqueeze reports two kinds of error bar: single-parameter and multi-parameter.

- The single-parameter error bar for a parameter b_j can be calculated analytically if we know the first and second derivatives of ϕ with respect to b_j . (The first derivative should, of course, be zero at the minimum). It is assumed here that, near the minimum, ϕ increases quadratically as each fitting parameter moves away from its optimum value. The single-parameter error bar is the calculated spread in the parameter which will result in a one-standard-deviation disagreement with the data, taken as a whole. That is, we estimate that if we re-measured the data many different times, we would get a fitted value of that parameter \pm the error bar 68% of the time. This assumes that all of the parameters are independent of each other. It turns out that increasing some b_j by its 1- σ uncertainty should have the effect of increasing ϕ by a factor:

$$\Phi_{\max} = \Phi_{\text{best}} \left(1 + \frac{1}{N - M} \right)$$

- The multi-parameter error bar for a parameter b_j takes into account the fact that parameters are not all independent, so that, if one parameter b_j is varied by a certain amount, the effect on the function can to some extent be “corrected for” by simultaneously changing a different parameter b_k . Thus, the multi-parameter error bars for a parameter are usually (but not always) somewhat greater than the single-parameter error bars, and sometimes much greater. *Datasqueeze* calculates the low and high ranges for a parameter by numerically testing how much the parameter can be changed before phi increases from ϕ_{best} to ϕ_{max} . This probably provides a better estimate of the true uncertainty in that parameter.

What Can Go Wrong

Beginners are sometimes misled by the fact that every scientific calculator includes a linear-regression feature into thinking that they can always trust the results of a least-squares fit. However, given the above considerations, it is not surprising that least-squares fits do not always yield expected (or correct) results. Here are some pitfalls to watch out for:

- **False Minima:** In the high-dimensional space of the parameters b_j , there may be many minima in ϕ . *Datasqueeze* just finds the one that is the closest to your starting parameters. This is analogous to the hiker who is lost on a volcano and finds his way to the bottom of the crater in the middle. He is at a local minimum in altitude, but nowhere near civilization. You should always make sure that your fit looks good--i.e., visually agrees with the data. If the outcome is really important to you, you should probably try a range of starting parameters and verify that you always end up at or near the same place.
- **Strongly-Coupled Parameters:** If two parameters are strongly coupled (i.e., if they do almost exactly the same thing to the function), the fit may not converge even after many iterations. Hopefully, when this happens, you will get an appropriate warning in the Message Area. Consider, for example, the following extreme case:

$$f(x) = Ax^{1.00000} + Bx^{1.00001}$$

Clearly, the two terms in this function do almost the same thing, and we would have to have a huge range in x before we could claim to have independently determined the values of A and B . You can check if two parameters are strongly coupled by clicking the Correlations button and looking at the values of the correlation coefficients.

- **Too Many Parameters:** This is related to the problem with strongly-coupled parameters. Suppose, for example, that we have a peak that is well described by a

single Gaussian function, with an amplitude, position, and width, but we believe that it really consists of two unresolved peaks. If we start with two Gaussian peaks and a total of six independent variables, the fit is almost certain to fail--if we are lucky, we will get a complaint about strongly correlated parameters, but it is also possible that the program will find a minimum based on noise in the data, with meaningless parameters. Such a fit might make sense if the parameters were restricted, for example by fixing the values of both widths such that there were only four fitting parameters, but extreme care must be taken.

- **Parameters That Have No Effect On The Fit:** Suppose that we have a diffraction peak that is “really” described by a function:

$$f(x) = Ae^{-((x-x_0)/d)^2}$$

with

$$A = 1000, x_0 = 0.1, d = 0.01$$

and we try to describe it with the correct function but with starting parameters

$$A = 1000, x_0 = 0.5, d = 0.001$$

This peak is far too sharp, and at the wrong position. The function is essentially zero every place that the data are nonzero. The fitting algorithm will never find its way to the right minimum. This is why it is crucial to have good starting values for all parameters.

Tricks for Avoiding Problems

- Make sure that your starting parameters are well chosen.
- Use different starting parameters and click on the Apply button several times until the curve agrees at least approximately with the data before trying to minimize anything.
- Check the Message Area after each fit to make sure that nothing went wrong, rather than just proceeding blindly and trusting the parameters that pop up.
- In general, vary as few parameters as possible.
- If χ^2 does not improve when you allow a parameter to vary, then that parameter is not having any statistically significant effect on the fit, and you should hold it fixed at some sensible value.
- If two parameters are very strongly correlated, at least one of them should be fixed.
- Start off by varying just one or two parameters, then allow more and more to vary. That way, you are less likely to have the program drift off into a parameter space of unphysical values.

•

Technical Details

Statistical Errors

Datasqueeze assumes that the number in each pixel represents the actual number of photons counted, so that the uncertainty is taken to be the square root of the value of that pixel. This is probably close to correct for wire and CCD detectors, less so for some other technologies such as image plates. Nevertheless, the overall effect is to weight intense data more than weak data, and final fitted results turn out to be remarkably insensitive to the exact algorithm chosen.

More precisely, if the “Sum” option is chosen in the Plot panel, then the independent variable y_i is the sum of all pixels that lie within that bin, and the error is taken to be $e_i = \sqrt{y_i}$ unless $y_i = 0$, in which case $e_i = 1$. If the “Average” option is chosen in the Plot panel, then for a data point with n_i pixels in the chosen range the independent variable is the sum of all pixel intensities divided by n_i , and the error is taken to be $e_i = \sqrt{y_i}/n_i$ unless $y_i = 0$, in which case $e_i = 1$.

In either case, the “weight” for least squares fits is one over the error, $w_i = 1/e_i$. Note that there is one place where this algorithm fails badly. If you have subtracted one data set from another (which is an allowed way to read in the data) then there may be many pixels for which the original data sets had lots of intensity but the difference pattern is small.

Least Squares Minimization Algorithm

Datasqueeze uses the Marquard nonlinear least-squares minimization algorithm (D. W. Marquardt, J. Soc. Ind. Appl. Math. II, **2**, 431-441 (1963)). The code was originally written in C, and was tested extensively at the Massachusetts Institute of Technology and the University of Pennsylvania. Due to the similarity between C and Java, a minimum number of changes were required to incorporate it into *Datasqueeze*.

One-Parameter Uncertainties

Suppose we have N data points y_i , each with uncertainty e_i and weight $w_i = 1/e_i$. At each point we have calculated a function f_i which depends on M independent parameters b_j . Then we define, as above,

$$\Phi = \sum_{i=1}^N \left(\frac{f(x_i) - y_i}{e_i} \right)^2$$

and

$$\chi^2 = \frac{\Phi}{N - M}$$

In the process of minimizing ϕ with respect to the parameters b_j , we will have calculated the derivative of f_i (at each point) with respect to b_j . Then we define

$$a_{jk} = \sum_{i=1}^N \left(\frac{\partial f_i}{\partial b_j} \right) \left(\frac{\partial f_i}{\partial b_k} \right) w_i^2$$

i.e.

$$a_{jj} = \sum_{i=1}^N \left(\frac{\partial f_i}{\partial b_j} \right)^2 w_i^2$$

Here the derivatives $(\partial f_i / \partial b_j)$ are of course partial derivatives; *Datasqueeze* calculates them numerically. It can then be shown (see, e.g., P. R. Bevington and D. K. Robinson, *Data Reduction and Error Analysis for the Physical Sciences*, Third Edition, McGraw Hill (2003)) that the uncertainty in parameter b_j is given by

$$\sigma_j = \sqrt{\frac{\chi^2}{a_{jj}}}$$

This is how the one-parameter uncertainties are calculated. Note that there is an implicit assumption that χ^2 is quadratic in each of *the* b_j ; i.e., we are using the second term in a Taylor expansion.

Multi-Parameter Uncertainties

The one-parameter uncertainties are calculated by taking a partial derivative of the function with respect to each of the b_j . This implicitly assumes that the parameters are uncorrelated. As discussed in Bevington, if ϕ_{best} is the value of ϕ obtained when all parameters have been optimized, then changing a given parameter to

$$b_j \rightarrow b_j \pm \sigma_j$$

should cause ϕ to increase to

$$\Phi_{\text{max}} = \Phi_{\text{best}} \left(1 + \frac{1}{N - M} \right)$$

assuming that no other parameters are varied. The multi-parameter error bar for a parameter b_j takes into account the fact that parameters are correlated, so that, if one parameter b_j is varied by a certain amount, the effect on the function can to some extent be “corrected for” by simultaneously changing a different parameter b_k . Thus, the multi-parameter error bars for a parameter are usually (but not always) somewhat greater than the single-parameter error bars, and sometimes much greater. *Datasqueeze* calculates the low and high ranges for a parameter by setting it to a sequence of different values, allowing all other parameters to vary, until ϕ increases from ϕ_{best} to ϕ_{max} .

Parameter Correlation Coefficients

The parameter correlation coefficients c_{jk} indicate how strongly two parameters are coupled. If $c_{jk}=1$ then parameters j and k do exactly the same thing to the model; if $c_{jk}=0$ then they are completely independent. The correlation coefficients are defined as follows: first we define (as above)

$$a_{jk} = \sum_{i=1}^N \left(\frac{\partial f_i}{\partial b_j} \right) \left(\frac{\partial f_i}{\partial b_k} \right) w_i^2$$

Then the c_{jk} are essentially normalized values of the a_{jk} . They are calculated approximately as follows:

$$c_{jk} = \frac{a_{jk}}{\sqrt{a_{jj}a_{kk}}}$$

except that care has to be taken if $a_{jj} \leq 0$. (In practice this means that there is a problem anyhow, because if one of the a_{jj} is zero then that parameter has no effect on the function).

Functions Provided

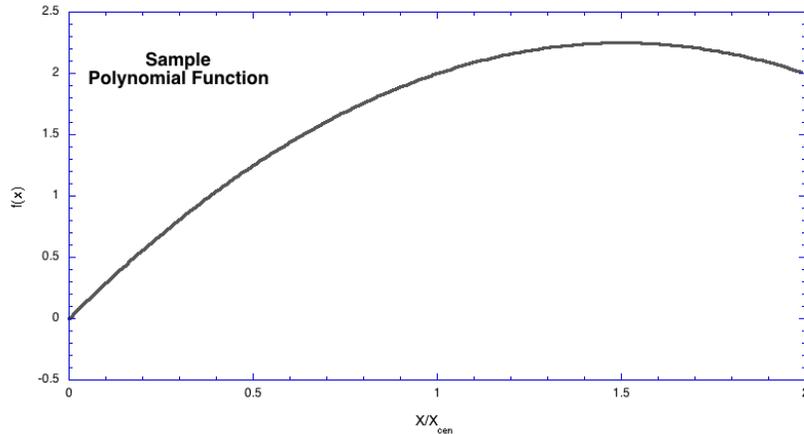
The following are brief descriptions of the functional forms currently available for least squares fits:

Polynomial:

A cubic polynomial. May be useful to describe a slowly varying background.

$$f = (\text{Const}) + (\text{Lin}) (x - \text{XC}) + (\text{Quad}) (x - \text{XC})^2 + (\text{Cub}) (x - \text{XC})^3$$

If $\text{abs}(\text{result})$ would be $>10^{30}$, returns 10^{30} . Should not vary all parameters simultaneously—normally hold XC fixed.



This model has four control cursors: one for XC and (Const), and one each for the linear, quadratic, and cubic terms. In Batch mode the model name is “Polynomial.”

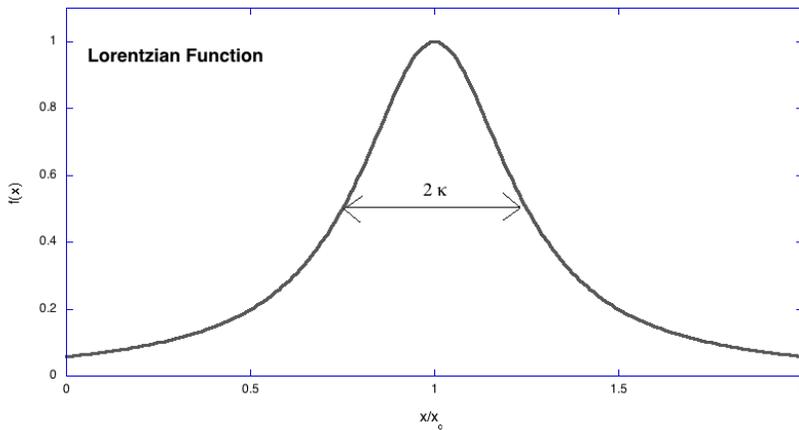
Lorentzian:

A Lorentzian peak function, often used to describe diffraction maxima from fluids.

$$f(x) = \frac{A\kappa^2}{(x - x_c)^2 + \kappa^2}$$

κ is the half-width at half maximum. The area under peak is $A\pi\kappa$. If $\kappa < 10^{-15}$, returns zero. If $\text{abs}(x_c) > 10^{15}$, returns zero. For structural analysis, the independent variable is normally q , not 2θ .

This model has two control cursors, one for the peak position and amplitude and one for κ . In Batch mode the model name is “Lorentzian.”

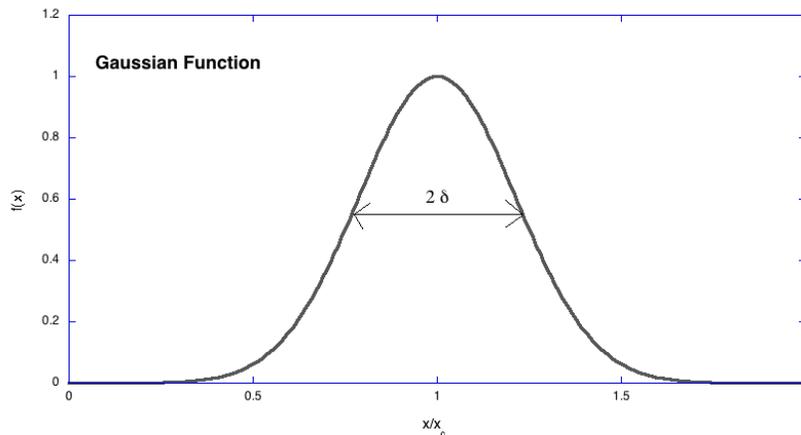


Gaussian

A Gaussian peak function, often used to describe Bragg peak shapes.

$$f(x) = A \exp\left(-\left(\frac{(x - x_c)\sqrt{\ln(2)}}{\delta}\right)^2\right)$$

δ is the half-width at half-maximum, x_c is position of center of peak. Area under peak is $A\delta\sqrt{\pi/\ln(2)}$. If $\text{abs}(\delta) < 10^{-10}$, returns zero. If $\text{abs}(x_c) > 10^{15}$, returns zero. If argument of exponential > 7 , returns zero. For structural analysis, the independent variable is normally q , not 2θ .



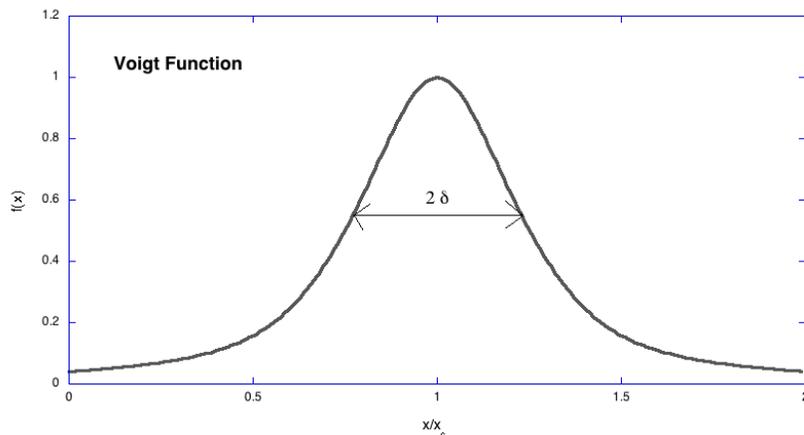
This model has two control cursors, one for the peak position and amplitude and one for δ . . In Batch mode the model name is “Gaussian”.”

Voigt

The model used in Datasqueeze is technically a “pseudo-Voigt” lineshape (the weighted average of a Lorentzian and a Gaussian) rather than a true Voigt lineshape (the convolution of a Lorentzian and a Gaussian, which takes substantially longer to calculate). This function is often used as an empirical lineshape for Bragg peaks.

$$z \equiv \frac{x - x_c}{\delta} \quad f = A \left(\frac{\alpha}{1 + z^2} + (1 - \alpha) \exp(-z^2 \ln(2)) \right)$$

δ is the half-width at half-maximum. Area under peak is $A\delta(\alpha\pi + (1 - \alpha)\sqrt{\pi/\ln(2)})$. See above descriptions of Gaussian and Lorentzian functions for overflow limits. Note that odd things may happen if $\alpha \ll 0$ or $\alpha \gg 1$. For structural analysis, the independent variable is normally q , not 2θ .



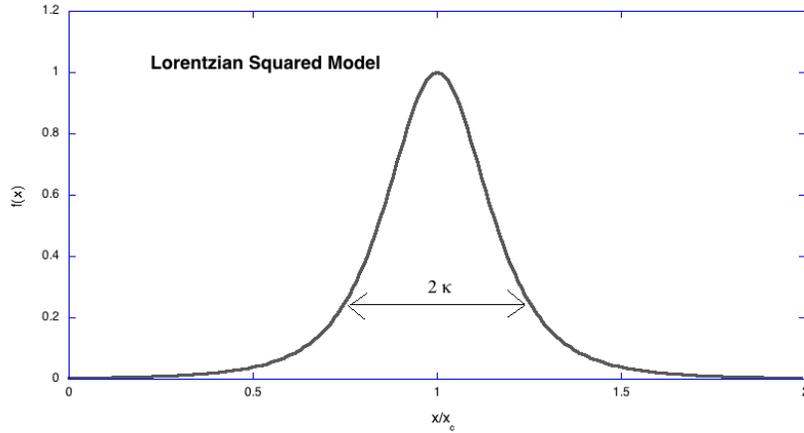
This model has three control cursors, one for the peak position and amplitude, one for δ , and one for α . In Batch mode the model name is “Voigt”.

Lorentzian Squared:

Sometimes useful to parametrize oddly-shaped peaks or beam zero scattering..

$$f = A \left(\frac{\kappa^2}{(x - x_c)^2 + \kappa^2} \right)^2$$

κ is the half-width at quarter maximum. Area under peak is $A\pi\kappa/2$. If $\kappa < 10^{-15}$, returns zero. If $\text{abs}(x_c) > 10^{15}$, returns zero. For structural analysis, the independent variable is normally q , not 2θ .



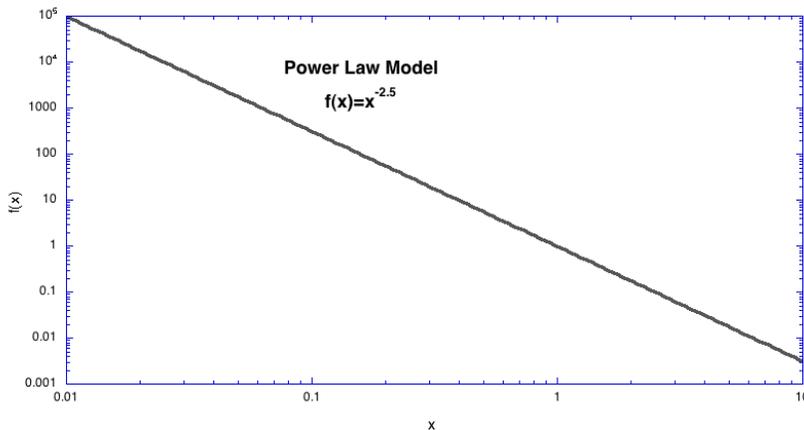
This model has two control cursors, one for the peak position and amplitude and one for κ . In Batch mode the model name is “Lorentzian^2”.

Power Law:

A power law function. May describe small angle scattering or fluctuation-limited peaks.

$$f = A|x - x_c|^\alpha$$

If argument diverges, returns $A \times 10^{20}$. Most often used with $x_c=0$. For structural analysis, the independent variable is normally q , not 2θ .



Note that setting the parameters (either with the parameter boxes or with cursors), and visually comparing the agreement between model and data, are much better done using a log-log plot than one with a linear scale.

This model has two control cursors, one for the amplitude, and one for the power law exponent. In Batch mode the model name is “Power Law”.

Radius of Gyration:

A Gaussian function describing small-angle scattering from a compact object with radius of gyration R_g .

$$f = Ae^{-(xR_g)^2/3}$$

To be meaningful, the independent variable should $x=q$, not 2θ . Note that setting the parameters (either with the parameter boxes or with cursors), and visually comparing the agreement between model and data, are much better done using a “Guinier plot” of $\log(\text{intensity})$ versus q^2 rather than one with a linear scale. In general, the plot scale has to be selected more carefully for this model than for some others. If the Guinier plot is not linear, you are probably outside the range of validity of the model.

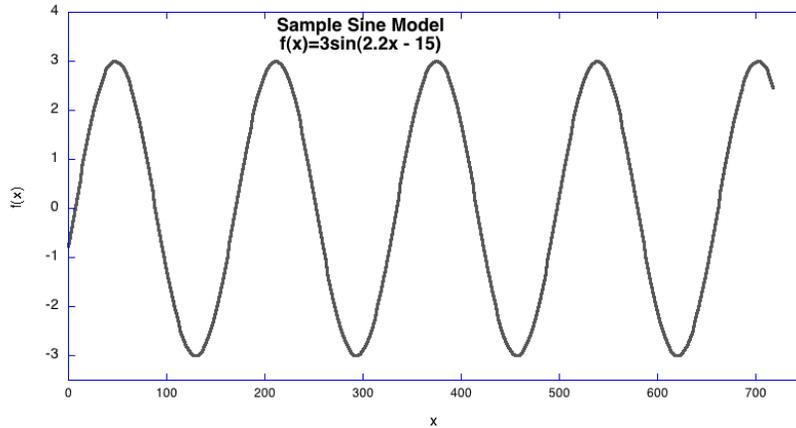
This model has one control cursor, which controls the amplitude and the radius of gyration. In Batch mode the model name is “Radius-Gyration”.

Sine:

A sine wave. Might be useful to describe azimuthal variation of a Bragg ring.

$$f = A \sin(\omega x + \phi)$$

Argument of sine is in degrees, not radians.



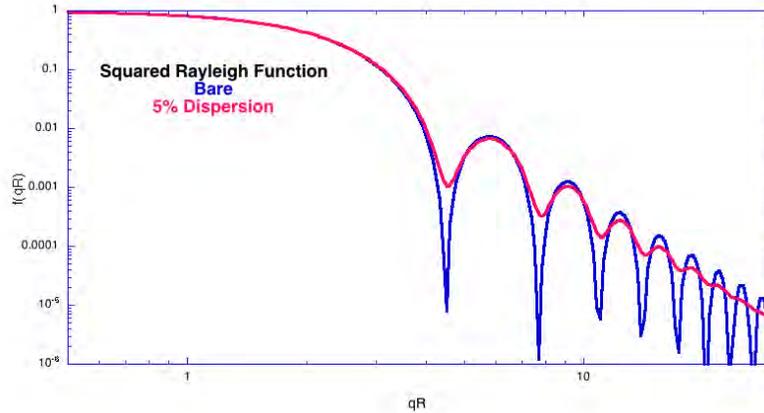
This model has two control cursors, one for the amplitude and phase, and one for the frequency ω . In Batch mode the model name is “Sine Wave”.

Rayleigh:

The Rayleigh function describes small-angle scattering from random dilute suspension of spheres, which possibly have polydisperse radii. Effective in version 2.2.4, the Gaussian distribution of radii was replaced by a log-normal distribution, which has a number of advantages. (It reduces to a Gaussian distribution in the limit of small dispersion, but never results in negative radii). The quoted value of σ is still the variance in the radius R . (See http://en.wikipedia.org/wiki/Log-normal_distribution for details of the log-normal distribution.) The bare function is:

$$f \equiv \left| \frac{3(\sin(Qr) - Qr \cos(Qr))}{(Qr)^3} \right|^2$$

For SAXS analysis the independent variable should be q , not 2θ .



This model has one control cursor, which determines the amplitude and mean radius. In Batch mode the model name “Rayleigh”.

Core-Shell:

The Core-Shell model is often used to describe nanoparticles with a spherical core and a spherical shell of a different electron density. If R_{core} is the radius of the core, R_{shell} the radius of the shell, ρ_{core} the electron density in the core, ρ_{shell} the electron density in the shell, and ρ_0 the density in the surrounding medium, then it is easily calculated that the scattered intensity should be proportional to

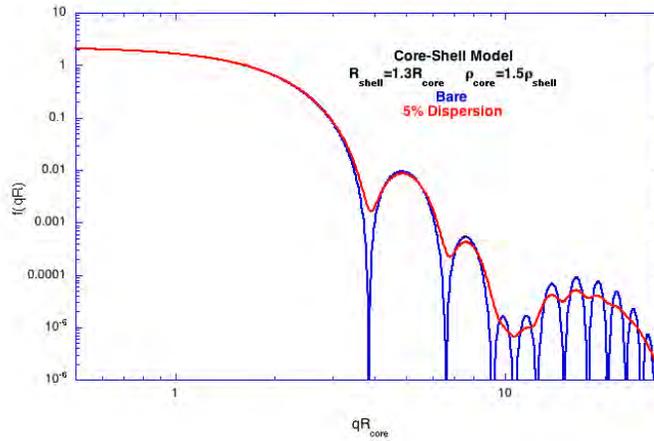
$$f = \left| (\rho_{shell} - \rho_0) R_{shell}^3 \Phi(qR_{shell}) + (\rho_{core} - \rho_{shell}) R_{core}^3 \Phi(qR_{core}) \right|^2$$

$$\Phi(x) \equiv \frac{3(\sin(x) - x \cos(x))}{x^3}$$

(Note that prior to version 3.0.4 the factors of R^3 were not included in the model).

The fitted densities are actually not the true electron densities, but rather the density differences between the scattering particle and the medium. That is, $R_{core} = (\rho_{core} - \rho_0)$ and $R_{shell} = (\rho_{shell} - \rho_0)$. Note that these densities are strongly coupled to the overall amplitude prefactor, so it is not possible to simultaneously fit “Ampl”, “Rcore”, and “Rshell”. The electron density of water is $\rho_0 = 0.334 \text{ e}^-/\text{Å}^3$.

As with the Rayleigh function, dispersion in the sphere radius is incorporated by numerically integrating over a log-normal distribution of radii. (If the dispersion is zero, just the bare function is returned). σ is taken to be the dispersion in R_{core} , with the ratio R_{shell} / R_{core} held fixed during the integration. Confusing and unphysical results may be obtained if $R_{shell} < R_{core}$ (but there is no problem in having one or both of the electron densities negative). For SAXS analysis the independent variable should be q , not 2θ .



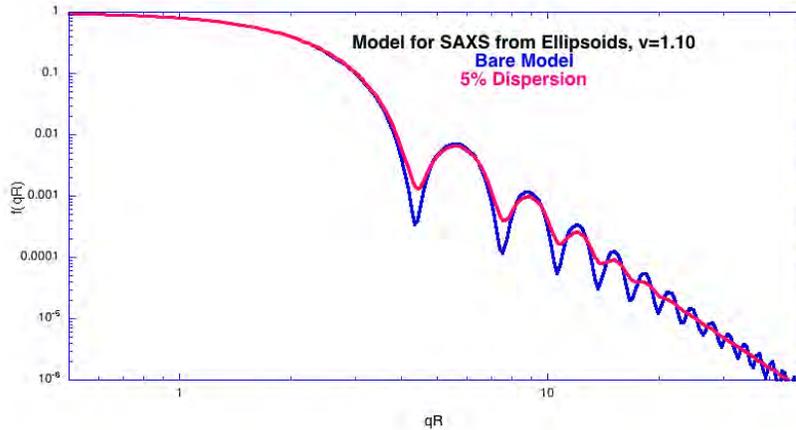
This model has one control cursor, which determines the amplitude and mean radius. In Batch mode the model name is “Core-Shell”.

Ellipsoid

This function describes small-angle scattering from a random dilute suspension of ellipsoids of revolution, with axes $2R$, $2R$, and $2vR$, where v is the aspect ratio. It is calculated by a numerical integration over spherical coordinates. Heterogeneity is incorporated by numerically integrating over a log-normal distribution of sphere radii. (If the dispersion is zero, just the bare function is returned). The “bare function” is

$$f = \int_0^{\pi/2} \Phi^2(qR\sqrt{\cos^2\theta + v^2\sin^2\theta}) \cos\theta d\theta$$

$$\Phi(x) \equiv \frac{3(\sin(x) - x\cos(x))}{(x)^3}$$



For SAXS analysis the independent variable should be q , not 2θ . Note that the aspect ratio and dispersion parameters are strongly coupled; for best results you should start with good guesses and vary as few parameters as possible, letting one additional parameter vary at a time.

See A. Guinier and G. Fournier, *Small-Angle Scattering of X-rays*, p. 19, Wiley & Sons (1955).

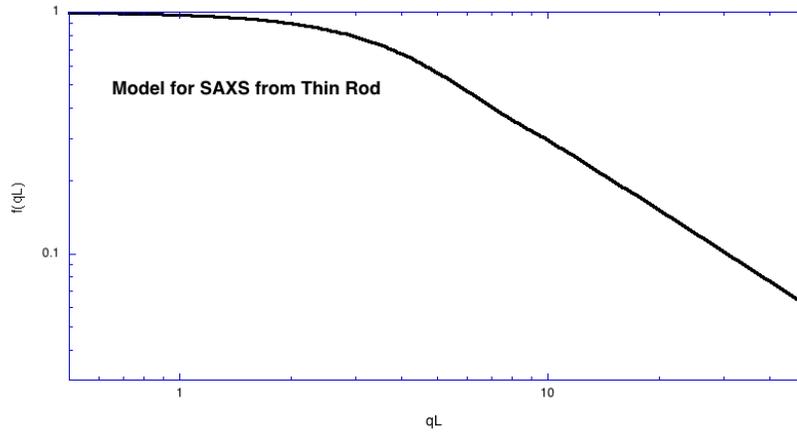
Since a multi-dimensional integral must be calculated at each point, it takes longer to evaluate this function than some of the others, and for this reason it was found impractical to incorporate control cursors. In Batch mode the model name is “Ellipsoid”.

Thin Rod

This function describes small-angle scattering from a random dilute suspension of rods of infinitesimal transverse dimension and length L . This function is so smooth that nothing is gained by Gaussian smoothing. The function is:

$$f = \frac{Si(qL)}{qL} - \frac{\sin^2(qL/2)}{(qL/2)^2}$$

$$Si(x) \equiv \int_0^x \frac{\sin(t)}{t} dt$$



For SAXS analysis the independent variable should be q , not 2θ .

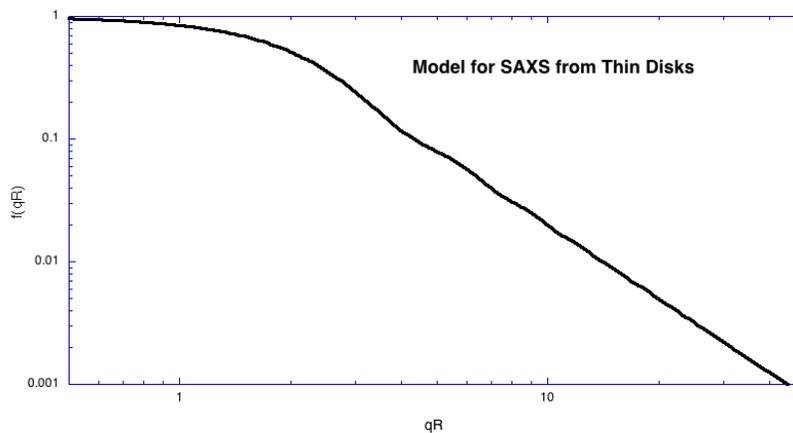
See A. Guinier and G. Fournier, *Small-Angle Scattering of X-rays*, p. 20, Wiley & Sons (1955).

This model has one control cursor, which determines the amplitude and rod length. In Batch mode the model name “ThinRod”.

Thin Disk

This function describes small-angle scattering from a random dilute suspension of flat disks of infinitesimal thickness and radius R . This function is so smooth that nothing is gained by Gaussian smoothing. The function is:

$$f = \frac{2}{(qR)^2} \left(1 - \frac{1}{qR} J_1(2qR) \right)$$



For SAXS analysis the independent variable should be q , not 2θ .

See A. Guinier and G. Fournier, *Small-Angle Scattering of X-rays*, p. 21, Wiley & Sons (1955).

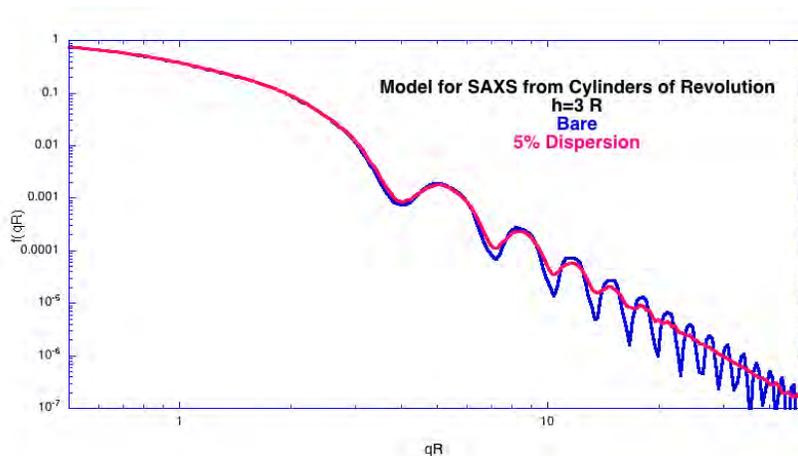
This model has one control cursor, which determines the amplitude and disk radius. In Batch mode the model name is “ThinDisk”.

Cylinder

This function describes small-angle scattering from a random dilute suspension of uniform-density cylinders (rods or disks) of radius R and height h . It is calculated via a numerical integration over spherical coordinates. Heterogeneity is incorporated by numerically integrating over a log-normal distribution of radii, with the ratio h/R kept constant. (If the dispersion is zero, just the bare function is returned). The “bare function” is

$$f = h^2 R^4 \int_0^{\pi/2} \frac{\sin^2((qh/2)\cos\theta)}{((qh/2)\cos\theta)^2} \frac{4J_1^2(qR\sin\theta)}{(qR\sin\theta)^2} \sin\theta d\theta$$

where $J_1(u)$ is the Bessel function of the first kind of order 1. (The prefactor of $h^2 R^4$ was added in version 3.0.4, and does not change anything except the fitted amplitude). Note that if the aspect ratio $v=h/R$ is either very large (resulting in a long thin rod) or very small (resulting in a thin disk), then the function is very smooth and little is changed by incorporating nonzero dispersion, and in these cases very similar results are expected from the Thin Disk or Thin Rod models, which can be calculated much more quickly. Oscillations are typically only observed if the aspect ratio is in the range $0.01 < v < 100$. Note also that parameters tend to be strongly coupled; for best results you should start with good guesses and vary as few parameters as possible, letting one additional parameter vary at a time.



For SAXS analysis the independent variable should be q , not 2θ .

See A. Guinier and G. Fournier, *Small-Angle Scattering of X-rays*, p. 19, Wiley & Sons (1955).

Since a multi-dimensional integral must be calculated at each point, it takes longer to evaluate this function than some of the others, and for this reason it was found impractical to incorporate control cursors. In Batch mode the model name is "Cylinder".

Coated Cylinder

This model extends the cylinder model to describe scattering from coated or functionalized cylinders, as might be found in assemblies of vesicles or nanoparticles. It is thus conceptually similar to the core-shell model often used to describe scattering from coated spheres. The physical model consists of the following:

- (a) A central core disk of radius R_{core} , height H_{core} , and electron density $\rho_{core} = Rho_{core} + \rho_0$, where ρ_0 is the electron density of the medium (often water, $\rho_{water} = 0.334 \text{ e}^-/\text{Å}^3$). The scattering amplitude from the core for vector components $q_z = q \cos\theta$, $q_r = q \sin\theta$ is:

$$s_1 = Rho_{core} h_{core} R_{core}^2 \frac{\sin((qh_{core}/2)\cos\theta)}{((qh_{core}/2)\cos\theta)} \frac{2J_1(qR_{core}\sin\theta)}{(qR_{core}\sin\theta)}$$

- (b) A ring of inner radius R_{core} , outer radius $R_{core} + T_{side}$, height H_{side} , and electron density $\rho_{side} = Rho_{side} + \rho_0$. The scattering amplitude from the side ring is

$$s_2 = Rho_{side} h_{side} \frac{\sin((qh_{side}/2)\cos\theta)}{((qh_{side}/2)\cos\theta)} \left(R_{side}^2 \frac{2J_1(qR_{core}\sin\theta)}{(qR_{core}\sin\theta)^2} - R_{core}^2 \frac{2J_1(qR_{core}\sin\theta)}{(qR_{core}\sin\theta)^2} \right)$$

If H_{side} is set to be negative it is forced to be equal to H_{core}

- (c)

Two "caps" of radius R_{capA} , centered on the core disk, that extend from $z = \pm H_{core}/2$ to $\pm(T_{cap} + H_{core}/2)$, with density $\rho_{capA} = Rho_{capA} + \rho_0$. The scattering amplitude from these caps is

$$s_3 = Rho_{capA} T_{capA} R_{capA}^2 \frac{\sin((qT_{capA}/2)\cos\theta)}{((qT_{capA}/2)\cos\theta)} \frac{2J_1(qR_{capA}\sin\theta)}{(qR_{capA}\sin\theta)^2} 2 \cos\left(q \cos\theta \frac{(T_{capA} + h_{core})}{2} \right)$$

If R_{capA} is set to be negative then it is forced to be equal to R_{core} .

(d)

Two more caps of radius R_{capB} and thickness T_{capB} , centered on the core disk, that extend from

$z = \pm H_{core}/2 + T_{capA}$ to $\pm(T_{capA} + H_{core}/2 + T_{capB})$, with density $\rho_{capB} = \rho_{capA} + \rho_0$. The scattering amplitude from these caps is

$$s_4 = \rho_{capB} T_{capB} R_{capB}^2 \frac{\sin((qT_{capB}/2)\cos\theta)}{((qT_{capB}/2)\cos\theta)} \frac{2J_1(qR_{capB}\sin\theta)}{(qR_{capB}\sin\theta)^2} 2\cos\left(q\cos\theta\left(\frac{h_{core} + T_{capB}}{2} + T_{capA}\right)\right)$$

If R_{capB} is set to be negative then it is forced to be equal to R_{core} .

The scattered intensity is then calculated by doing a spherical average over the square of the summed amplitudes:

$$f = \int_0^{\pi/2} (s_1 + s_2 + s_3 + s_4)^2 \sin\theta d\theta$$

Note also that parameters tend to be strongly coupled; for best results you should start with good guesses and vary as few parameters as possible, letting one additional parameter vary at a time. It is not possible to vary the amplitude prefactor and all of the densities at the same time. For SAXS analysis the independent variable should be q , not 2θ .

Since a multi-dimensional integral must be calculated at each point, it takes longer to evaluate this function than some of the others, and for this reason it was found impractical to incorporate control cursors. In Batch mode the model name is "Coated Cylinder".

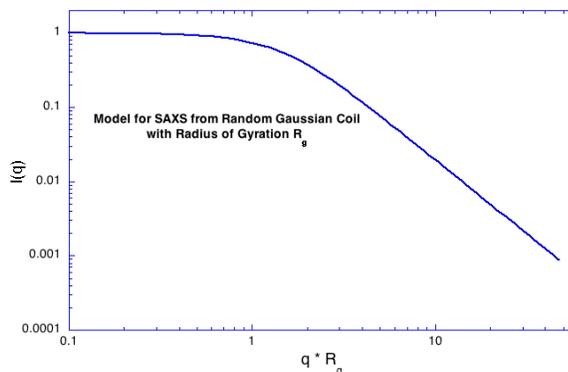
Gaussian Coil

This function describes small-angle scattering from a flexible polymer chain which is not self-avoiding and obeys Gaussian statistics. The function is

$$f = 2\left(\frac{\exp(-u) + u - 1}{u^2}\right)$$
$$u \equiv q^2 R_g^2$$

where R_g is the radius of gyration.

For SAXS analysis the independent variable should be q , not 2θ .



For the original calculation see P. Debye, *J. Phys. Colloid. Chem.* **51**, 18-23 (1947).

This model has one control cursor, which determines the amplitude and radius of gyration. In Batch mode the model name is “GaussianCoil”.

Fractal Aggregate

This function describes a model for small-angle scattering from fractal aggregates of spheres. The bare function is

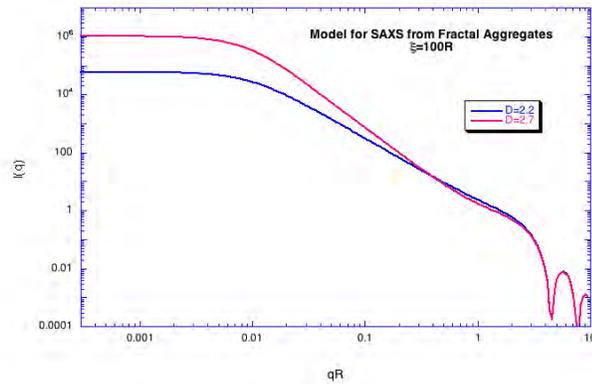
$$f(q) = S(q) |F(q)|^2$$

$$S(q) = 1 + \frac{1}{(qR)^D} \frac{D\Gamma(D-1)}{\left[1 + 1/(q^2\xi^2)\right]^{(D-1)/2}} \sin\left[(D-1)\tan^{-1}(q\xi)\right]$$

$$F(q) = 3 \frac{\sin(qR) - qR \cos(qR)}{(qR)^3}$$

Here D is the fractal dimension of the system, R is the radius of the individual spheres, and ξ represents the characteristic distance above which the mass distribution is no longer described by a fractal law. Γ is the Gamma function. Note that the model assumes $2 \leq D \leq 3$ and $\xi > R$; unexpected and unphysical results may be obtained if these conditions are not met.

For SAXS analysis the independent variable should be q , not 2θ .



See J. Teixeira, *J. Appl. Cryst.* **21**, 781-785 (1988) and also J. S. Pederson in *Neutrons, X-rays, and Light: Scattering Methods Applied to Soft Condensed Matter*, P. Lindner and Th. Zemb eds, Elsevier (2002), pp. 391-420.

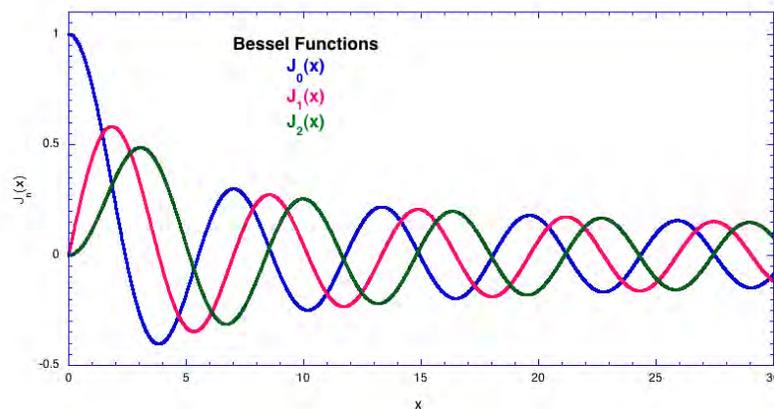
This model has two control cursors, one determining the amplitude and R and the other determining ξ . In Batch mode the model name is “Fractal Aggregate”.

Bessel Function:

The Bessel function is sometimes found in the analysis of helical diffraction patterns. Datasqueeze allows you to use any integer Bessel function of the first kind. The function is

$$f \equiv A j_n(qR)$$

Here n is the order of the Bessel function, rounded to the nearest integer; it should not be varied.



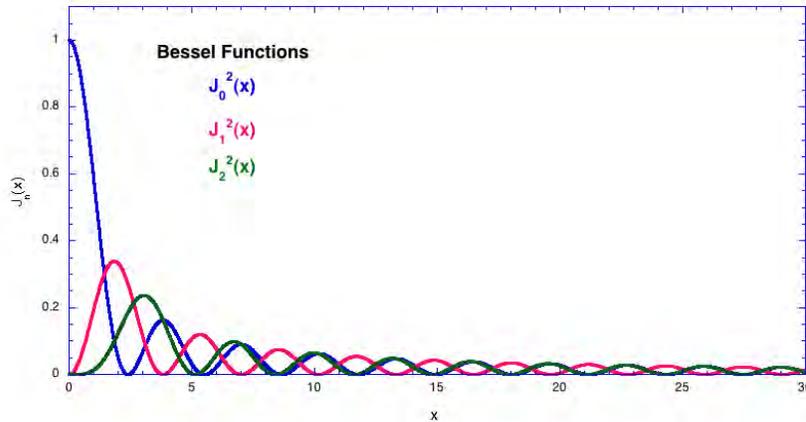
This model has one control cursor, which determines the amplitude and radius R. In Batch mode the model name is “Bessel”.

Bessel Function Squared:

Same as the Bessel function (above), but squared.

$$f \equiv A|j_n(qR)|^2$$

Here n is the order of the Bessel function, rounded to the nearest integer; it should not be varied.



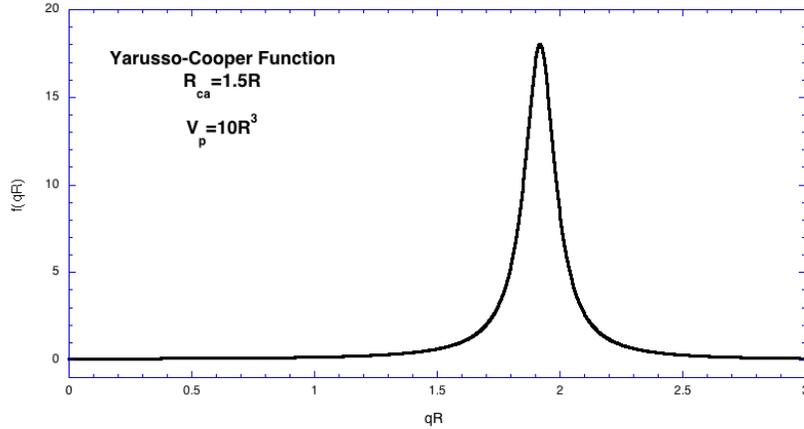
This model has one control cursor, which determines the amplitude and radius R. In Batch mode the model name is “Bessel^2”.

Yarusso-Cooper:

The Yarusso-Cooper function is sometimes used to describe scattering from ionomers or micelles. It incorporates the Rayleigh form factor with hard-sphere correlations.

$$f = \frac{A\Phi^2(qR_1)}{1 + \frac{8V_{ca}}{V_p}\Phi(2qRCa)} \quad \Phi(u) \equiv \frac{3(\sin(u) - u\cos(u))}{u^3} \quad V_{ca} \equiv \frac{4\pi R_{Ca}^3}{3}$$

Here R_l is the radius of the scattering object (assumed to be a sphere of uniform electron density), R_{CA} is the distance of closest approach of two scatterers, and V_p is the mean volume per particle of a scatterer. Note that unphysical results will be obtained if $V_{CA} \geq V_p$. In fact, any packing fraction greater than about 0.75 is unphysical, and the YC model should work best in a regime even more dilute than that.



For SAXS analysis the independent variable should be q , not 2θ .

See D. J. Yarusso and S. L. Cooper, *Macromolecules* **16**, 1871 (1983).

This model has two control cursors, one for the amplitude and radius R_l , and one for the volume V_p . In Batch mode the model name is “Yarusso-Cooper”.

Kinning-Thomas:

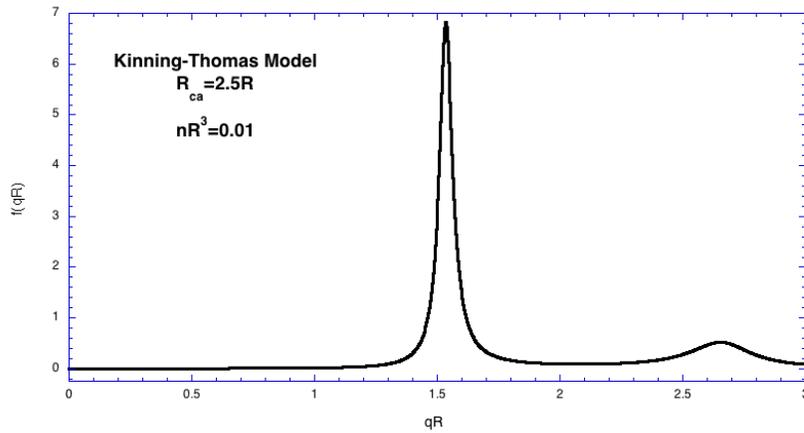
The Kinning-Thomas function is another model used to describe scattering from ionomers or micelles. It incorporates a Rayleigh form factor with Percus-Yevick correlations. It represents an analytical solution of the Percus-Yevick-Rayleigh model (below).

$$f = \frac{A\Phi^2(qR)}{1 + 24\eta(G(A)/A)} \quad \Phi(u) \equiv \frac{3(\sin(u) - u\cos(u))}{u^3} \quad \eta = \frac{4\pi R_{CA}^3 n}{3}$$

$$A = 2QR_{CA} \quad \alpha = \frac{(1 + 2\eta)^2}{(1 - \eta)^4} \quad \beta = \frac{-6\eta(1 + \eta/2)^2}{(1 - \eta)^4} \quad \gamma = \frac{(\eta/2)(1 + 2\eta)^2}{(1 - \eta)^4}$$

$$G(A) = \frac{\alpha}{A^2}(\sin A - A \cos A) + \frac{\beta}{A^3}(2A \sin A + (2 - A^2)\cos A - 2) \\ + \frac{\gamma}{A^5}(-A^4 + 4[(3A^2 - 6)\cos A + (A^3 - 6A)\sin A + 6])$$

Here R is the radius of the scattering object (assumed to be a sphere of uniform electron density), R_{ca} is the distance of closest approach of two scatterers, and n is the volume density of scatterers. As with the Yarusso-Cooper model, any packing fraction greater than about $\eta = 0.75$ is unphysical, and the KT model should work best in a regime even more dilute than that.



For SAXS analysis the independent variable should be q , not 2θ .

See D. J. Kinning and E. L. Thomas, *Macromolecules* **17**, 1712 (1984).

This model has two control cursors, one for the amplitude and radius R , and one for the packing fraction η . In Batch mode the model name is "Kinning-Thomas".

Percus-Yevick:

The Percus-Yevick structure factor describes the scattering from point-like objects with a hard sphere repulsive potential. (No form factor is included).

$$f = \frac{A}{1 - nc(Qd)}$$

$$c(Qd) = -4\pi d^3 \int_0^1 ds s^2 \frac{\sin sQd}{sQd} (\alpha + \beta s + \gamma s^3)$$

$$d = 2r = \text{diameter}$$

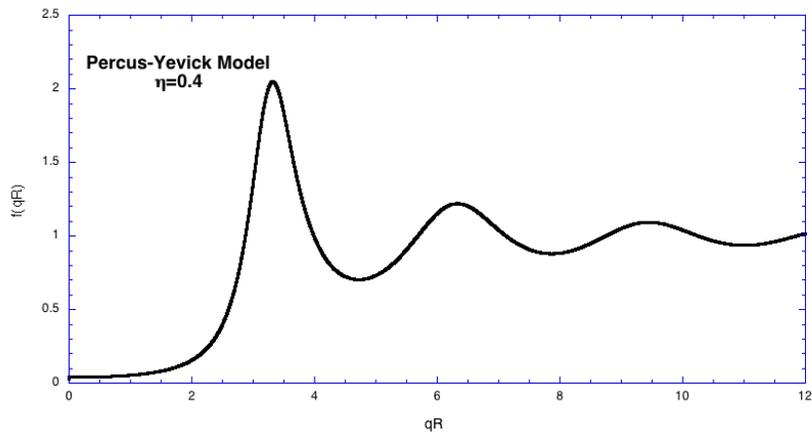
η = volume fraction of scatterers

$$n = \frac{6\eta}{\pi d^3} = \text{number density of scatterers}$$

$$\alpha = \frac{(1+2\eta)^2}{(1-\eta)^4}$$

$$\beta = \frac{-6\eta(1+\eta/2)^2}{(1-\eta)^4}$$

$$\gamma = \frac{(\eta/2)(1+2\eta)^2}{(1-\eta)^4}$$



Note that $\eta > 0.75$ is an unphysically high packing density. For SAXS analysis the independent variable should be q , not 2θ .

See J. K. Percus and G. J. Yevick, *Phys. Rev.* **110**, 1 (1958); N. W. Ashcroft and J. Lekner, *Phys. Rev.* **145**, 83 (1966).

This model has two control cursors, one for the amplitude and radius R_1 , and one for the packing fraction η . In Batch mode the model name "Percus-Yevick".

Scripted (“Batch”) Mode

Overview

Datasqueeze provides an interface for running in a “batch” mode, otherwise known as “scripting” or “running from a file”. The scripting file is a simple ASCII file that can be created with any text editor. It should be raw text--that is, it should not contain any formatting commands (bold, etc.) such as those produced by Word or most other text editors. It is suggested that the file have a suffix .txt, since this will indicate to Windows and Macintosh systems that this is a plain text file, but in fact *Datasqueeze* itself doesn't care. This feature is primarily useful if you want to process a number of data files in the same way, without visually examining each one. For example, you might want to read in a set of files, and for each one save a false-color image and an ASCII version of the plot.

Each line starts with the > character followed by a keyword and some optional arguments. There are batch versions of most of the GUI-based actions. There are not batch versions of actions such as *Examine* which are intrinsically interactive.

Although the examples given in this manual employ numbers in the American format (e.g., 1,234.56) the number format should reflect whatever language you have chosen as your preference on your computer. A short script might consist of something like the following:

```
>READFILE O “myfile.raw”
>CEN 497.0 519.0
>SAVEIMAGE “myfile.jpg” JPEG
>RECALCIMAGE
>PLOT false false false Q 0.01 0.092 0.0010 CHI 3.0 170.0
>SAVEPLOT “myplot.txt” TAB
```

The first line tells the program to open the data file myfile.raw which is in Bruker-Siemens format. The second line tells it to save the false color image in jpeg format. The third line sets the center position. The fourth line redisplay the image. The fifth line sets up the parameters for a linear plot, and draws the plot. The last line saves the ASCII data from that plot.

Once you have set up your data file, you can invoke it by clicking the *Read From Command File* menu item under Batch. Once the commands have been executed (or an error encountered) the program returns to graphical mode.

You can also create a log of the commands you run while in graphical mode by selecting *Enable Logging Commands* under the Batch file. The log will be continuously updated until you exit the program or select *Disable Logging commands*. It is suggested again that you give your file a .txt suffix. The file thus created is identical in format to an input file for Batch mode. Thus, you can create a sample template by running in graphical mode with logging turned on, and then edit that file to create a sequence of commands that will operate on many different files. In fact, this is the suggested way of creating such files, since otherwise it is hard to be sure that you have correctly specified the full path name for files appropriate to your system.

You probably will, however, want to edit the log file before just running it, because otherwise it is likely to run somewhat slowly. For example, in graphical mode the image is redrawn every time some important change is made, such as zooming in or changing the center. This results in many time-consuming calls to RECALCIMAGE. When running in a batch or background mode you probably want to call this command at most once per file.

The remainder of this document gives a synopsis of each available command, in alphabetical order.

Commands

ADDMOVIEFRAME	
Purpose:	Add the current false color image to the buffer of movie frames
Usage	> ADDMOVIEFRAME
Parameters:	None
Example	> ADDMOVIEFRAME
See also:	INITIALIZEMOVIE, PLAYMOVIE, SAVEMOVIE

APPENDTOFILE	
Purpose:	Enable or disable appending data upon readin
Usage	> APPENDTOFILE tf
Parameters:	tf (boolean): True to append new data to existing data
Example	> APPENDTOFILE true

APPLYFIT	
Purpose:	Apply least-squares model to plot without actually doing a fit
Usage	> APPLYFIT show_submodel
Parameters:	show_submodel (boolean): True to show sub-models in plot
Example	> APPLYFIT true

APPLYMASK	
Purpose:	Apply masks that have just been defined
Usage	> APPLYMASK
Parameters:	none
Example	> APPLYMASK

AXISRANGE	
Purpose:	Reset scale for a plot axis
Usage	>AXISRANGE axisname startval endval deltaval
Parameters:	axisname (string): Name of axis (X or Y)
	startval (double): Lower limit value
	endval (double): Upper limit value
	deltaval (double): Tick increment
Example	>AXISRANGE X 0.01 0.09 0.005
Comment:	This command only makes sense after a call to PLOT.

AZIMUTH	
Purpose:	Set azimuth for detector tilt correction
Usage	>AZIMUTH az
Parameters:	az (double): Angle of azimuth (in degrees)
Example	>AZIMUTH 82.3

BEAMCEN		
Purpose:	Set center of beam zero	
Usage	> BEAMCEN xc yc	
Parameters:	xc (double):	X-pixel of beam zero (may be fractional)
	yc (double):	Y-pixel of beam zero (may be fractional)
Example	> BEAMCEN 515.2 428.7	

CALCSTATISTICS		
Purpose:	Calculate pixel statistics	
Usage	> CALCSTATISTICS userranges	
Parameters:	userranges (boolean):	True to use user-entered pixel ranges, false for default ranges
	Example	> CALCSTATISTICS false
See also	The SELECTREGION command allows one to select the region of interest without doing a plot, while the PLOT command selects a region and also does the plot. One or the other should be called before the first call to CALCSTATISTICS	

CHIOFFSET		
Purpose:	Set the definition of chi=0 for azimuthal plots (useful for oriented patterns)	
Usage	> CHIOFFSET val	
Parameters:	Val (double):	Set orientation of new equator in degrees
Example	> CHIOFFSET 39.7	

COMMENT		
Purpose:	Comment in batch file--does nothing	
Usage	> COMMENT	
Parameters:		none
Example	> COMMENT	

DETECTORTWOTHETA		
Purpose:	Set detector arm 2theta (wax mode)	
Usage	> DETECTORTWOTHETA tt	
Parameters:	az (double):	Detector arm 2 theta (wax mode) (in degrees)
Example	> DETECTORTWOTHETA 15.73	

DEZING		
Purpose:	Clean up bad pixels	
Usage	> DEZING	
Parameters:		none
Example	> DEZING	

DIAMETER		
Purpose:	Set detector diameter. This is the side-to-side width of the detector, in cm, not the diagonal span.	
Usage	> DIAMETER da	
Parameters:	da (double):	Detector diameter in cm
Example	> DIAMETER 8.53	

DOFIT		
Purpose:	Do least-squares fit of model to data	
Usage	> DOFIT mpeb_set show_submodel	
Parameters:	mpeb_set (boolean):	True to calculate multi-parameter error bars
	show_submodel (boolean):	True to show sub-models in plot
Example	> DOFIT false true	

EXPORTPLOT		
Purpose:	Save plot data in one of several standard formats	
Usage	> EXPORTPLOT filename format [anodetype]	
Parameters:	filename (string):	Name of file, in quotes
	format (string):	File format; valid formats are CPI, CSV, or GSAS , PLV, DBWS, ASCIIPROFILE, ASCII2COLUMN
	anodetype (string)	Anode type; allowed values are Cu, Mo, Au, or Synch
Examples	> EXPORTPLOT "myplot.cpi" CPI AU > EXPORTPLOT "myplot.csv" CSV	

FILESWAP	(DEPRECATED)
Purpose:	This command no longer has any effect.

FITVSQ		
Purpose:	Force least squares fit to use Q as independent parameter even if plotting parameter was 2θ or Q^2	
Usage	> FITVSQ fq	
Parameters:	fq (boolean):	True to force fit vs. Q, false to use plot variable
Example	> FITVSQ true	

FLIPFIVERTICAL		
Purpose:	Reflect image about horizontal axis	
Usage	> FLIPFIVERTICAL	
Example	> FLIPFIVERTICAL	

FLIPHORIZONTAL		
Purpose:	Reflect image about vertical axis	
Usage	FLIPHORIZONTAL	
Example	> FLIPHORIZONTAL	

FORCEGRID		
Purpose:	Force grid to use user-defined values	
Usage	FORCEGRID fg	
Parameters:	fg (Boolean)	True to use user-defined values for grid on the image, false for auto-selection
Example	> FORCEGRID true	
See also:	The GRIDINCREMENTS command sets the actual grid values that will be used. It should be called after FORCEGRID is set to true.	

GRIDINCREMENTS		
Purpose:	Set steps for grid on top of false color image. Negative increments autoscale that variable	
Usage	GRIDINCREMENTS qdelta tthdelta chidelta qxdelta qydelta	
Parameters:	qdelta (double):	Radial increment for q-chi polar grid
	tthdelta (double):	Radial increment for 2 θ -chi polar grid
	chidelta (double):	Azimuthal increment polar plot
	qxdelta (double):	Horizontal increment for Cartesian grid
	qydelta (double):	Vertical increment for Cartesian grid
Examples	>GRIDINCREMENTS 0.02 -1 5.0 -1 -1 >GRIDINCREMENTS -1 -1 10.0 0.025 0.025	
See also:	The FORCEGRID command determines whether the grid will be autoscaled or not. GRIDINCREMENTS has no effect unless preceded by FORCEGRID true	

HORIZSYM		
Purpose:	Fold the data so the image has right-left symmetry about the beam center.	
Usage	HORIZSYM	
Examples	> HORIZSYM	
See also:	INVERTSYM, VERTSYM	

INITIALIZEMOVIE	
Purpose:	Initialize a new movie of false color images.
Usage	INITIALIZEMOVIE
Examples	> INITIALIZEMOVIE
See also:	ADDMOVIEFRAME, PLAYMOVIE,SAVEMOVIE

IMAGEDISPLAYENABLE	
Purpose:	Enable or disable display of false color image
Usage	> IMAGEDISPLAYENABLE truefalse
Parameters:	truefalse (boolean): True to show false color image, false otherwise
Example	> IMAGEDISPLAYENABLE false

IMAGEAUTORECALCENABLE	
Purpose:	Enable or disable autoresize of false color image upon file open
Usage	> IMAGEAUTORECALCENABLE truefalse
Parameters:	truefalse (boolean): True to autoresize false color image, false otherwise
Example	> IMAGEAUTORECALCENABLE false

IMPORT1D		
Purpose:	Import one-dimensional data	
Usage	IMPORT1D filename variable_name column_delimiter uncert xlog ylog [lambda] [nskip] [nread]	
Parameters:	filename (string):	Name of input file
	variable_name (string):	Name of independent variable: should be Q, Q ² , 2THETA, CHI, QX, or QY
	column_delimiter (string):	Column delimiter: should be TAB, SPACE, or COMMA
	uncert (boolean):	True if file includes a third column for uncertainties, false otherwise
	xlog (boolean):	True if horizontal axis should be plotted on a log scale, false otherwise
	ylog (boolean):	True if vertical axis should be plotted on a log scale, false otherwise
	Lambda (double)	Wavelength (angstroms).
	nskip (integer)	Number of lines at the beginning of the file to skip (default 0)
	nread (integer)	Total number of lines to read after the skipped lines (default entire file)
Example	>IMPORT1D "C:\sample data\input.txt" Q TAB true false false 1.54 1 75	

INVERTSYM		
Purpose:	Fold the data so the image has inversion symmetry about the beam center.	
Usage	INVERTSYM	
Examples	> INVERTSYM	
See also:	HORIZSYM, VERTSYM	

LAMBDA		
Purpose:	Set wavelength	
Usage	> LAMBDA la	
Parameters:	la (double):	Wavelength (angstroms)
Example	>LAMBDA 1.54	

LENGTH		
--------	--	--

Purpose:	Set sample-detector distance	
Usage	> LENGTH le	
Parameters:	le (double):	Sample-detector distance in cm
Example	>LENGTH 123.5	

LINEARSCALE		
Purpose:	Set linear intensity scale for image	
Usage	> LINEARSCALE	
Parameters:		none
Example	>LINEARSCALE	

LOGSCALE		
Purpose:	Set logarithmic intensity scale for image	
Usage	> LOGSCALE	
Parameters:		none
Example	> LOGSCALE	

MAKE3DIMAGE		
Purpose:	Make 3D Image of the data	
Usage	> MAKE3DIMAGE sz	
Parameters:	sz (integer):	Size of image, in pixels. (512, 768, or 1024)
Example	> MAKE3DIMAGE 512	

MASK		
Purpose:	Define a new Mask	
Usage	> MASK maskname index v1 [v2 v3 v]	
Parameters:	maskname (string)	Type of mask being defined. Should be one of MINXMASK, MINYMASK, MAXXMASK, MAXYMASK, ROWMASK, COLUMNMASK, PIXELMASK BOXMASK, INSIDECIRCLEMASK, OUTSIDECIRCLEMASK.
	Index (integer)	Index of the mask being defined (0, 1, 2, up to NMASK)
	v1, v2, v3, v4 (double)	Values of parameters defining this mask

		(between 1 and 4 values)
Example	> MASK MINXMASK 0 5	
	> MASK BOXMASK 6 255 176 823 954	

MAXVALUE		
Purpose:	Set data value corresponding to maximum in color scale	
Usage	> MAXVALUE maxval	
Parameters:	maxval (double):	Maximum intensity
Example	>MAXVALUE 550.	

MINVALUE		
Purpose:	Set data value corresponding to minimum in color scale	
Usage	> MINVALUE minval	
Parameters:	minval (double):	Minimum intensity
Example	> MINVALUE 2.7	

MULTIPLYSINTheta		
Purpose:	Enable or disable multiplication by Lorentz factor of $\sin(\theta)$	
Usage	> MULTIPLYSINTheta tf	
Parameters:	tf (boolean):	True to multiply by $\sin(\theta)$, false otherwise
Example	> MULTIPLYSINTheta true	

MULTIPLYSINTTheta		
Purpose:	Enable or disable multiplication by Lorentz factor of $\sin(2\theta)$	
Usage	> MULTIPLYSINTTheta tf	
Parameters:	tf (boolean):	True to multiply by $\sin(2\theta)$, false otherwise
Example	> MULTIPLYSINTTheta false	

NCONTOUR		
Purpose:	Set number of contour lines to use	
Usage	> NCONTOUR nc	
Parameters:	nc (integer):	Number of contours, $20 \geq nc \geq 1$

Example	> NCONTOUR 14
---------	---------------

NMASK	
Purpose:	Set number of masks to use for data filtering
Usage	> NMASK nm
Parameters:	nm (integer): Number of masks, $nm \geq 4$
Example	> NMASK 7

NMODELS	
Purpose:	Set number of submodels to add for least-square fit
Usage	> NMODELS nm
Parameters:	nm (integer): Number of contours, $1 \leq nm \leq 2$
Example	> NMODELS 3

NUMSTATBINS	
Purpose:	Set number of bins to use for statistics
Usage	> NUMSTATBINS bin_count
Parameters:	bin_count (integer): Number of bins, $1 \leq bin_count \leq 20$
Example	> NUMSTATBINS 15

PIXELSIZE	
Purpose:	Set detector pixel size
Usage	> PIXELSIZE pi
Parameters:	pi (double): Detector pixel size in micrometers
Example	> PIXELSIZE 20.5

PLAYMOVIE		
Purpose:	Play movie of false color images.	
Usage	> PLAYMOVIE [latency [ncycles]]	
Parameters:	latency (integer):	Time between frames in msec, $1 \leq \text{bin latency count} \leq 20$. Default value is 100
	ncycles (integer):	Number of time to play movie, $1 \leq \text{bin ncycles count} \leq 32767$. Default value is 5
Examples	>PLAYMOVIE 200 6 > PLAYMOVIE 50 > PLAYMOVIE	
See also:	ADDMOVIEFRAME, INITIALIZEMOVIE	

PLOT		
Purpose:	Make a 1D plot	
Usage	> PLOT xlog ylog sum indvar mini maxi deltai consvar minc maxc [deltac [expon]]	
Parameters:	xlog (boolean):	True for logarithmic horizontal axis, false for linear axis
	ylog (boolean):	True for logarithmic vertical axis, false for linear axis
	sum (boolean):	True if we are to sum (or integrate) over pixels in range, false if we are to average over pixels
	indvar (string):	Name of variable over which we are to plot--possibilities are Q, CHI, TTHETA, QX, QY, or LINEX.
	mini (double):	Minimum value of independent variable
	maxi (double):	Maximum value of independent variable
	delti (double):	Plotting increment of independent variable
	consvar (string):	Name of variable over which we are to integrate --possibilities are Q, CHI, ELLIPSE, TTHETA, QX, QY, or LINEY
	minc (double):	Minimum value of constraint variable
	maxc (double):	Maximum value of constraint variable
	delti (double):	Plotting increment of independent variable--only used if plotting along a line or with ellipse
	Expon (integer):	Exponent for independent variable
Examples	>PLOT false false true Q 0.01 0.05 0.001 CHI 10 350	

	>PLOT true false false LINEX 563.0 755.0 0.01 LINEY 231.0 403.0 0.01 >PLOT false false true CHI 10 350 2.5 Q 0.01 0.05 0.001 2 >Plot false false false CHI 0.000 360.000 2.500 ELLIPSE 1.244 1.474 1.480
--	--

PRINT					
Purpose:	Print a summary page				
Usage	> PRINT printtype[user text]				
Parameters:	<table border="1"> <tr> <td>printtype (string):</td> <td>Kind of printout we are making; choices are "SUMMARY", "IMAGE", or "PLOT"</td> </tr> <tr> <td>user text (string):</td> <td>User string, prepended to automatically generated information about this file</td> </tr> </table>	printtype (string):	Kind of printout we are making; choices are "SUMMARY", "IMAGE", or "PLOT"	user text (string):	User string, prepended to automatically generated information about this file
printtype (string):	Kind of printout we are making; choices are "SUMMARY", "IMAGE", or "PLOT"				
user text (string):	User string, prepended to automatically generated information about this file				
Examples	>PRINT IMAGE >PRINT SUMMARY "T=250K, B=0.3 T\nSample 45"				
Comment	This is not purely Batch mode, since on most platforms a print command calls up a dialog box. So this command cannot be run in "unattended" mode. Linux/Unix printing not currently supported.				

Q-RANGE			
Purpose:	Set q-scale for detector		
Usage	> Q-RANGE qr		
Parameters:	<table border="1"> <tr> <td>qr (double):</td> <td>Maximum horizontal extent of detector in inverse angstroms</td> </tr> </table>	qr (double):	Maximum horizontal extent of detector in inverse angstroms
qr (double):	Maximum horizontal extent of detector in inverse angstroms		
Example	>Q-RANGE 0.128		

READFILE															
Purpose:	Read in a data file. Replaces READIN command.														
Usage	>READFILE filename [dettype usefilecalib addval multval imno]														
Parameters	<table border="1"> <tr> <td>filename (integer):</td> <td>which file in the list this is. If it is zero, any previous data in memory are overwritten. If it is greater than zero, the new data are added to previous data.</td> </tr> <tr> <td>filename (string):</td> <td>full name of the file we are reading in, enclosed in quotes</td> </tr> <tr> <td>dettype (string):</td> <td>Name of detector type, enclosed in quotes Currently supported types: "ADSC_Quantum", "Binary-little-endian", "Binary-big-endian", "CrysAlis", "Bruker-Siemens", "Bruker-Nonius KCD", "Ditabis IPC", "Esperanto", "ESRF", "Fuji_IP", "Gatan DM3", "GrayScale Image", "MAR_CCD", "MAR_IP (300)", "MAR_IP (345)", "MAR_IP (PCK)", "Molecular Metrology", "Nonius DIP", "Rigaku_Raxis", "RaxisIV++", "Roper-Princeton", "SBIG", "Tiff", "XDR", "Unknown." The last one does its best to choose among the other available choices. If no name is specified, will attempt to use the last format used.</td> </tr> <tr> <td>usefilecalib (boolean):</td> <td>true to use calibration parameters from file metadata, false to use user-determined values. Default is the value specified on the Calibration Parameter Source buttons on the File panel.</td> </tr> <tr> <td>addval (double):</td> <td>quantity to add to each pixel , default 0</td> </tr> <tr> <td>multval (double):</td> <td>quantity by which we should multiply each pixel, default 1</td> </tr> <tr> <td>imno (integer):</td> <td>Image to select from within the file (default -1 adds them all up)</td> </tr> </table>	filename (integer):	which file in the list this is. If it is zero, any previous data in memory are overwritten. If it is greater than zero, the new data are added to previous data.	filename (string):	full name of the file we are reading in, enclosed in quotes	dettype (string):	Name of detector type, enclosed in quotes Currently supported types: "ADSC_Quantum", "Binary-little-endian", "Binary-big-endian", "CrysAlis", "Bruker-Siemens", "Bruker-Nonius KCD", "Ditabis IPC", "Esperanto", "ESRF", "Fuji_IP", "Gatan DM3", "GrayScale Image", "MAR_CCD", "MAR_IP (300)", "MAR_IP (345)", "MAR_IP (PCK)", "Molecular Metrology", "Nonius DIP", "Rigaku_Raxis", "RaxisIV++", "Roper-Princeton", "SBIG", "Tiff", "XDR", "Unknown." The last one does its best to choose among the other available choices. If no name is specified, will attempt to use the last format used.	usefilecalib (boolean):	true to use calibration parameters from file metadata, false to use user-determined values. Default is the value specified on the Calibration Parameter Source buttons on the File panel.	addval (double):	quantity to add to each pixel , default 0	multval (double):	quantity by which we should multiply each pixel, default 1	imno (integer):	Image to select from within the file (default -1 adds them all up)
filename (integer):	which file in the list this is. If it is zero, any previous data in memory are overwritten. If it is greater than zero, the new data are added to previous data.														
filename (string):	full name of the file we are reading in, enclosed in quotes														
dettype (string):	Name of detector type, enclosed in quotes Currently supported types: "ADSC_Quantum", "Binary-little-endian", "Binary-big-endian", "CrysAlis", "Bruker-Siemens", "Bruker-Nonius KCD", "Ditabis IPC", "Esperanto", "ESRF", "Fuji_IP", "Gatan DM3", "GrayScale Image", "MAR_CCD", "MAR_IP (300)", "MAR_IP (345)", "MAR_IP (PCK)", "Molecular Metrology", "Nonius DIP", "Rigaku_Raxis", "RaxisIV++", "Roper-Princeton", "SBIG", "Tiff", "XDR", "Unknown." The last one does its best to choose among the other available choices. If no name is specified, will attempt to use the last format used.														
usefilecalib (boolean):	true to use calibration parameters from file metadata, false to use user-determined values. Default is the value specified on the Calibration Parameter Source buttons on the File panel.														
addval (double):	quantity to add to each pixel , default 0														
multval (double):	quantity by which we should multiply each pixel, default 1														
imno (integer):	Image to select from within the file (default -1 adds them all up)														
Examples	<pre>>READFILE 0 "C:\sample data\test.raw" "Bruker-Siemens" false 0.0 1.0 -1 >READFILE 1 "C:\sample data\test.raw" "Bruker-Siemens" >READFILE 0 "C:\sample data\test.raw"</pre>														

READIN	(DEPRECATED)
--------	--------------

Purpose:	Read in a data file. Still works, but superseded by READFILE	
Usage	>READIN filename restore accstats filename dettype addval multval imno	
Parameters	filename (integer):	which file in the list this is. If it is zero, any previous data in memory are overwritten. If it is greater than zero, the new data are added to previous data.
	restore (boolean):	true if we are to keep same values for xcen, etc., as before, false if we are to use values from file
	accstats (boolean):	Has no effect
	filename (string):	name of the file we are reading in, enclosed in quotes
	dettype (string):	Name of detector type, enclosed in quotes Currently supported types: "ADSC_Quantum", "Binary-big-endian", "Binary-little-endian", "Bruker-Nonius KCD", "Ditabis IPC", "Esperanto", "ESRF", "Fuji_IP", "Gatan DM3", "GrayScale Image", "MAR_CCD", "MAR_IP (300)", "MAR_IP (345)", "MAR_IP (PCK)", "Molecular Metrology", "Nonius DIP", "Rigaku_Raxis", "RaxisIV++", "Roper-Princeton", "SBIG", "Tiff", "XDR", "Unknown." The last one does its best to choose among the other available choices.
	addval (double):	quantity to add to each pixel
	multval (double):	quantity by which we should multiply each pixel
	imno (integer):	Image to select from within the file (default -1 adds them all up)
Example	>READIN 0 1 false "C:\sample data\test.raw" "Bruker-Siemens" 0.0 1.0 -1	

RECALCIMAGE		
Purpose:	Force recalculation of image from possibly changed data	
Usage	> RECALCIMAGE	
Parameters:	None	
Example	>RECALCIMAGE	

RESTOREDEFAULTIMAGEVALUES	
---------------------------	--

Purpose:	Restore false color image settings to default values
Usage	> RESTOREDEFAULTIMAGEVALUES
Parameters	None
Example	>RESTOREDEFAULTIMAGEVALUES

RESTOREFITENVIRONMENT	
Purpose:	Restore saved current fit environment
Usage	> RESTOREFITENVIRONMENT filename
Parameters:	filename (string): Name of file, in quotes
Example	> RESTOREFITENVIRONMENT "fitenvironment.txt"

RETRIEVEINSTRUMENTPARAMETERS	
Purpose:	Retrieve (import) calibration parameters saved in file
Usage	> RETRIEVEINSTRUMENTPARAMETERS filename
Parameters:	filename (string): Name of file, in quotes
Example	> RETRIEVEINSTRUMENTPARAMETERS "parameters.txt"

ROTATE	
Purpose:	Rotate image 90 degrees
Usage	> ROTATE
Parameters:	None
Example	> ROTATE

SAVEDATA	
----------	--

Purpose:	Save 2D data in an ASCII file	
Usage	> SAVEDATA filename separator	
Parameters:	filename (string):	Name of file, in quotes
	Separator (string)	Separator between columns; valid separators are TAB, COLUMN, or SPACE
Example	> SAVEDATA "mydata.txt" TAB	

SAVEDATAASTIFF		
Purpose:	Save 2D data in a tiff file	
Usage	> SAVEDATAASTIFF filename	
Parameters:	filename (string):	Name of file, in quotes
Example	> SAVEDATAASTIFF "mydata.tif"	

SAVEFITENVIRONMENT		
Purpose:	Save current fit environment	
Usage	> SAVEFITENVIRONMENT filename	
Parameters:	filename (string):	Name of file, in quotes
Example	> SAVEFITENVIRONMENT "fitenvironment.txt"	

SAVEFITPARAMETERS		
Purpose:	Save fit parameters in an ASCII file	
Usage	> SAVEFITPARAMETERS filename rowcol	
Parameters:	filename (string):	Name of file, in quotes
	rowcol (boolean):	True to save data with one row per parameter, false to save data with one column per parameter in plot
Example	> SAVEFITPARAMETERS "fitresults.txt" true	
Comment:	You should always call CONSTRUCTMODEL first, as it is not guaranteed that a model exists until you do this.	

SAVE3DIMAGE		
Purpose:	Save pseudo-3D image	
Usage	> SAVE3DIMAGE filename	
Parameters:	filename (string):	Name of file, in quotes
	filetype (string):	graphics format Currently supported types: "JPEG", "PNG", "TIFF"
Example	>SAVE3DIMAGE "file.jpg" JPEG	

SAVEIMAGE		
Purpose:	Save false color image in graphics format	
Usage	> SAVEIMAGE filename filetype [resolution]	
Parameters:	filename (string):	Name of file, in quotes
	filetype (string):	graphics format Currently supported types: "JPEG", "PNG", "TIFF"
	resolution (integer)	Saved image will have resolution x resolution pixels. Default set by "Saved Image Resolution" button
Examples	>SAVEIMAGE "file.jpg" JPEG >SAVEIMAGE "file.tif" TIFF 1024	

SAVEINSTRUMENTPARAMETERS	
Purpose:	Save calibration parameters in file
Usage	> SAVEINSTRUMENTPARAMETERS filename
Parameters:	filename (string): Name of file, in quotes
Example	> SAVEINSTRUMENTPARAMETERS "param.txt"

SAVEMOVIE	
Purpose:	Save movie as a .avi file.
Usage	> SAVEMOVIE latency filename
Parameters:	latency (integer): Time between frames in msec, $1 \leq \text{bin}$ latency count ≤ 20 . Default value is 100
	filename (string): Name of file, in quotes
Example	
See also:	ADDMOVIEFRAME, INITIALIZEMOVIE, PLAYMOVIE

SAVEOVERLAY	
Purpose:	Save grid and selected region overlays when copying or writing image
Usage	> SAVEOVERLAY dosave
Parameters:	dosave (Boolean): True to save
Example	> SAVEOVERLAY true

SAVEPLOT	
Purpose:	Save plot data in an ASCII file
Usage	> SAVEPLOT filename separator
Parameters:	filename (string): Name of file, in quotes
	separator (string): Separator between columns: valid separators are TAB, COMMA, or SPACE
Example	> SAVEPLOT "myplot.txt" COMMA

SAVEPLOTIMAGE	
---------------	--

Purpose:	Save plot image in graphics format	
Usage	> SAVEPLOTIMAGE filename filetype	
Parameters:	filename (string):	Name of file, in quotes
	filetype (string):	graphics format Currently supported types: "JPEG", "PNG, TIFF
Example	> SAVEPLOTIMAGE "file.jpg" JPEG	

SAVESTATISTICS		
Purpose:	Save file statistics in an ASCII file	
Usage	> SAVESTATISTICS filename	
Parameters:	filename (string):	Name of file, in quotes
Example	> SAVESTATISTICS "mystats.txt"	
Comment:	You should always call CALCSTATISTICS first, as SAVESTATISTICS is not guaranteed to bring them up to date.	

SELECTREGION		
Purpose:	Select plotting region without doing plot (Mostly in conjunction with statistics calculations).	
Usage	> SELECTREGION indvar mini maxi consvar minc maxc	
Parameters:	indvar (string):	Name of variable over which we are to plot-- possibilities are Q, CHI, TTHETA, QX, QY, or LINEX.
	mini (double):	Minimum value of independent variable
	maxi (double):	Maximum value of independent variable
	consvar (string):	Name of variable over which we are to integrate --possibilities are Q, CHI, TTHETA, QX, QY, or LINEY
	minc (double):	Minimum value of constraint variable
	maxc (double):	Maximum value of constraint variable
Example	> SELECTREGION Q 0.02 0.05 CHI 10 270	

SETANTIALIAS		
Purpose:	Enable or disable antialiasing correction for line plots	
Usage	> SETANTIALIAS ai	
Parameters:	ai (boolean):	True for antialiasing (nicer but slow), false otherwise
Example	> SETANTIALIAS true	

SETCOSINEAB		
Purpose:	Set parameters for polarization correction to intensity, $I' = I / (A + B \cos^2 2\theta)$	
Usage	> SETCOSINEAB ca cb	
Parameters:	ca (double)	'A' term
	cb (double)	'B' term
Example	> SETCOSINEAB 0.5 0.25	

SETINTENSITYEXPONENT		
Purpose:	Set exponent n for $x^n \times I$ plots	
Usage	> SETINTENSITYEXPONENT exponent	
Parameters:	exponent (integer):	Integer between 0 and 4
Example	> SETINTENSITYEXPONENT 2	

SETCOLOR		
Purpose:	Choose color scheme for false color image	
Usage	> SETCOLOR ncolor	
Parameters:	ncolor (integer):	Index of color chosen, $0 \leq \text{ncolor} \leq 7$
Example	> SETCOLOR 3	

SETCONTOUR		
Purpose:	Set fields for a particular contour line	
Usage	> SETCONTOUR nc val redval greenval blueval	
Parameters:	nc (integer):	Index of contour line to be updated, $0 \leq \text{nc} \leq 20$
	val (integer):	Intensity associated with this contour line
	redval (integer):	Red component of the RGB composition of this color, $0 \leq \text{redval} \leq 255$
	greenval (integer):	Green component of the RGB composition of this color, $0 \leq \text{greenval} \leq 255$
	blueval (integer):	Blue component of the RGB composition of this color, $0 \leq \text{blueval} \leq 255$
Example	> SETCONTOUR 3 237 0 125 253	

SETFITPARAMETER									
Purpose:	Set value and fitting status for a fit parameter								
Usage	> SETFITPARAMETER model_number parameter_number allow_fit [value]								
Parameters:	<table border="1"> <tr> <td>model_number (integer):</td> <td>Which fit model this parameter this goes with, $1 \leq \text{model_number} \leq 6$</td> </tr> <tr> <td>parameter_number (integer):</td> <td>Which parameter within the model, typically, $1 \leq \text{parameter_number} \leq 3$</td> </tr> <tr> <td>allow_fit (boolean):</td> <td>Whether this parameter should be allowed to vary (true to vary)</td> </tr> <tr> <td>value (double):</td> <td>New value of this parameter</td> </tr> </table>	model_number (integer):	Which fit model this parameter this goes with, $1 \leq \text{model_number} \leq 6$	parameter_number (integer):	Which parameter within the model, typically, $1 \leq \text{parameter_number} \leq 3$	allow_fit (boolean):	Whether this parameter should be allowed to vary (true to vary)	value (double):	New value of this parameter
model_number (integer):	Which fit model this parameter this goes with, $1 \leq \text{model_number} \leq 6$								
parameter_number (integer):	Which parameter within the model, typically, $1 \leq \text{parameter_number} \leq 3$								
allow_fit (boolean):	Whether this parameter should be allowed to vary (true to vary)								
value (double):	New value of this parameter								
Examples	> SETFITPARAMETER 1 2 true > SETFITPARAMETER 2 1 false 45.23								

SETGRID			
Purpose:	Determine type of grid to put o an image		
Usage	> SETGRID gridtype		
Parameters:	<table border="1"> <tr> <td>gridtype (string):</td> <td>Type of grid. Currently supported types: "NONE", "Q-CHI", "2THETA-CHI", "QX-QY"</td> </tr> </table>	gridtype (string):	Type of grid. Currently supported types: "NONE", "Q-CHI", "2THETA-CHI", "QX-QY"
gridtype (string):	Type of grid. Currently supported types: "NONE", "Q-CHI", "2THETA-CHI", "QX-QY"		
Example	> SETGRID Q-CHI		

SETLINEWIDTH			
Purpose:	Set width of lines in plot		
Usage	> SETLINEWIDTH lw		
Parameters:	<table border="1"> <tr> <td>lw (double):</td> <td>Width of line, in pixels, $0.1 \leq \text{lw} \leq 10.0$. Default is 1.0.</td> </tr> </table>	lw (double):	Width of line, in pixels, $0.1 \leq \text{lw} \leq 10.0$. Default is 1.0.
lw (double):	Width of line, in pixels, $0.1 \leq \text{lw} \leq 10.0$. Default is 1.0.		
Example	> SETLINEWIDTH 2.5		

SETMARKERINDEX		
Purpose:	Set index corresponding to size of marker in plots	
Usage	> SETMARKERINDEX index	
Parameters:	index (integer):	Which choice of radius. Possible values are: 0 -- Auto 1 -- Line Only 2 -- Radius = 1.0 3 -- Radius = 1.5 4 -- Radius = 2.0 5 -- Radius = 2.5 6 -- Radius = 3.0 7 -- Radius = 3.5 8 -- Radius = 4.0 9 -- Radius = 4.5 10-- Radius = 5.0
Example	> SETMARKERINDEX 2	

SETMARKERTYPE		
Purpose:	Set the type of marker used for plotting	
Usage	> SETMARKERTYPE mt	
Parameters:	mt (string):	Name of marker type, enclosed in quotes Currently supported types: "Filled Circles", "Open Circles", "Filled Squares", "Open Squares", "Filled Upwards Triangles", "Open Upwards Triangles", "Filled Downwards Triangles", "Open Downwards Triangles", "Filled Diamonds", "Open Diamonds", "Filled Pentagons", "Open Pentagons"
Example	> SETMARKERTYPE "Open Diamonds"	

SETMODEL		
Purpose:	Set the kind of submodel we are using	
Usage	>SETMODEL model_number model_name	
Parameters:	model_number (integer):	which model in the list this is (1 ≤ n_model ≤ 6)
	model_name (string):	Name of this parameter, enclosed in quotes Currently supported models are “Polynomial”, “Lorentzian”, “Gaussian”, “Voigt”, “Lorentzian^2”, “Power Law”, “Radius-Gyration”, “Sine Wave”, “Rayleigh”, “Core-Shell”, “Ellipsoid”, “Cylinder”, “ThinRod”, “ThinDisk”, ”Coated Cylinder”, “GaussianCoil”, “Fractal Aggregate”, “Bessel”, “Bessel^2”, “Yarusso-Cooper”, "Percus-Yevick" and "Kinning-Thomas".
Example	> SETMODEL 3 “Gaussian”	

SETPLOTGRID		
Purpose:	Determines the grid overlay on linear plots	
Usage	> SETPLOTGRID grid_type	
Parameters:	grid_type (string)	Grid type: currently allowed values are NONE, COARSE, or FINE
Example	> SETPLOTGRID “FINE”	

SETPLOTGRIDCOLOR		
Purpose:	Determines the color of plot grid	
Usage	> SETPLOTGRIDCOLOR grid_color	
Parameters:	grid_color (string)	Grid color: currently allowed values are GRAY (default), RED, GREEN, BLUE, ORANGE, CYAN, OR magenta
Example	> SETPLOTGRIDCOLOR “BLUE”	

SETSTATBIN		
Purpose:	Set range for a particular statistics bin	
Usage	> SETSTATBIN bin_number lowval highval	
Parameters:	bin_number (integer):	Index of bin to be updated, $0 \leq \text{bin_number} \leq 20$
	lowval (integer):	Low pixel counts for this bin
	highval (integer):	High pixel counts for this bin
Example	> SETSTATBIN 10 127 238	

SHOWEB		
Purpose:	Enable or disable display of error bars in line plot	
Usage	> SHOWEB eb	
Parameters:	eb (boolean):	True to display error bars
Example	> SHOWEB true	

SHOWTEXT		
Purpose:	Enable or disable display of text in line plot	
Usage	> SHOWTEXT st	
Parameters:	st (boolean):	True to display text
Example	> SHOWEB true	

SMOOTH		
Purpose:	Smooth data	
Usage	> SMOOTH	
Parameters:	None	
Example	>SMOOTH	
TILT		
Purpose:	Set tilt for detector tilt correction	
Usage	> TILT ti	
Parameters:	ti (double):	Tilt of detector away from normal to beam (in degrees)
Example	>TILT 4.7	

TWOTHETARANGE		
---------------	--	--

Purpose:	Set maximum detector 2theta	
Usage	> TWOTHETARANGE de	
Parameters:	de (double):	Set 2theta at maximum horizontal extent of detector (in degrees)
Example	> TWOTHETARANGE 0.235	

VERTSYM		
Purpose:	Fold the data so the image has top-bottom symmetry about the beam center.	
Usage	VERTSYM	
Examples	> VERTSYM	
See also:	HORIZSYM, INVERTSYM	

WAXSMODE		
Purpose:	Enable or disable wide angle mode	
Usage	> WAXSMODE tf	
Parameters:	tf (boolean):	True for wide-angle mode, false for small-angle mode
Example	> WAXSMODE true	

WRITEFILENAME		
Purpose:	Write file name on the false color image	
Usage	> WRITEFILENAME wf	
Parameters:	wf (Boolean)	True to draw name on image
Example	> WRITEFILENAME true	

ZOOM		
Purpose:	Zoom in to image	
Usage	> ZOOM zv	
Parameters:	zv (integer):	Zoom value: must be -1 (zoom out 2), 1, 2, 4, 8, 16, or 32
Example	> ZOOM 2	

ZOOMCEN		
Purpose:	Set pixel at center of zoomed-in image	
Usage	> ZOOMCEN xc yc	
Parameters:	xc (double):	X-pixel of center (may be fractional)
	yc (double):	Y-pixel of center (may be fractional)
Example	> ZOOMCEN 65.0 237	

2DFFT		
Purpose:	Perform Fourier Transform of data	
Usage	> 2DFFT zv filter	
Parameters:	zv	Magnification: must be -1 (zoom <i>out</i> 2), 1, 2, 4, 8, 16, or 32
	filter	Type of filter, should be one of “Hamming”, “Pseudo-Lorentzian”, “Gaussian”, or “None”
Example	> 2DFFT 8 Hamming	

Formats used in Batch Commands

String

String variables are used for things like filenames. If they contain any spaces they must be enclosed in quotes using the double-quote “ character (not two single quotes). They must not contain the characters '>', '%', or ‘‘’. They can however contain the newline character '\n' if it is not part of a file name.

Examples:

test.raw

“C:\sample data\test.raw”

“Descriptive material for printout.\nTwo lines.”

Bad examples:

C:\sample data\test.raw

"test.raw" (*looks OK but was made with pairs of single quotes rather than double quotes*)

“/users/junk->junk”xxx”/test%3.dat” (*includes multiple illegal characters*)

Integer

Integer variables should have no commas or decimal points. Hexadecimal or octal variables are not supported.

Examples:

1
1500

Bad examples:

1.0
1,500
0431
0ae15c6

Double

Double variables should have no commas. Decimal points or exponentials are optional. We use the American, rather than the European, convention for decimals, i.e. a full stop (".") rather than a comma (",").

Examples

1.0
1.0e+03
5

Bad example:

1,000

Boolean

Boolean variables express things that are either true or false. They should be represented either by the word "true" (without quotes) or the word "false". Upper case is allowable. Any word other than "true" is assumed to represent "false."

Version History

Changes from version 3.0.3

- Improvements in stability and performance.
- Improved appearance of logarithmic plots.
- Bigger selection of format options when intensity vs. 2theta plot is exported
- Added option in 1D import to limit number of lines read in.
- Modified the Core-shell and Cylinder fitting models. The different densities are explicitly made into fitting factors (although you cannot modify all densities and the amplitude simultaneously), the different size parameters (radii, thickness) are made into explicit fitting parameters rather than using ratios, and factors of R^3 are correctly incorporated in the relative amplitudes of the different components.
- Implemented a “Coated Cylinder” fitting model, which is conceptually similar to the core-shell model for spheres.
- Added option to save and subsequently retrieve the fit environment (submodels used and parameters) in a file.

Changes from version 3.0.2

- Improvements in stability and performance
- Copying of false color and plot image on Macintosh OSX deprecated (you can still save these as image files)

Changes from version 3.0.1

- Window positions are saved between sessions.
- Added “Save Plot Image” option to “Process Multiple Files” sequence.
- Maximum number of points in a plot increased in recognition of increased frame sizes of many detectors.
- Corrected minor bugs in manual calibration feature.
- Improved options for importing 1D (x-y) data.
- Added support for Ganesha enhancements to Tiff format.
- Continued improvements in performance and stability.
- Full compliance with Macintosh 10.9 (“Mavericks”)

Changes from version 3.0.0

- Added support for Agilent Esperanto format.
- Improved support for Princeton-Roper format.

Changes from version 2.2

- Continued improvements in performance and stability.

- Added fitting models for “Radius of Gyration” and scattering from fractal aggregates,
- Added the option of appending data to existing data, so that in principle an arbitrary number of input files can be summed.
- Added option (in Advanced Plot Features) to correct for Lorentz and Polarization factors.
- Verified compatibility with Windows 8, and now consistent with latest security features in Macintosh OS 10.8 (“Mountain Lion”)
- An important change in the plots are made: the recorded value of the independent variable was previously at the beginning of the integration interval, and is now in the center. For example, if a user decided to do a plot from $q=0.1$ to $q=0.2$ in steps of 0.01, data points were previously recorded as being at $q=0.1, 0.2, 0.3,$ etc., with the point at $q=0.1$ indicating the integrated value from 0.1 to 0.2. The same intensity is now recorded as being measured at $q=0.15$. This change should have very little effect on plots recorded with a small plot increment, but may change values if the plot increments are large.
- Opening multiple data files in simultaneous separate windows was deprecated. This originally seemed like a good idea, but frequently led to confusion at both the software and user levels.
- Added support Princeton/Roper files with floating point data, and for CBF data files
- Added support for saving indexed lines, saving movies, exporting raw data as a tiff file, making saved image resolution user-selectable, and saving false color images as tiff files
- Added tools for making pseudo-3D images of the data and quickly converting between different units for describing diffraction data, namely $Q, S, D, \theta,$ or $2-\theta$.

Changes from version 2.1

- Added support for SBIG and Oxford Instruments CrysAlis format and improved support for extended Tiff files, ESRF files, and ADSC files.
- Added fitting models for cylinders, thin rods, thin disks, ellipsoids of revolution, Gaussian coils, the Kinning-Thomas model, and “core-shell” spheres.
- Added option for analyzing data in a “wide-angle” mode, in which the detector sits on a 2-theta arm that may have been rotated about a large angle away from the beam zero. Added more options for exporting data.
- Implemented Fourier-transformed image of data
- A variety of bug fixes, stability improvements, and enhancements to the user interface.
- Some older Batch features were deprecated.

Frequently Asked Questions

Installation

- *Can I run my copy of Datasqueeze on more than one computer?* You can run the Trial version on as many computers as you want. Your license allows you to run the full version on at most 5 computers. The spirit of this rule is that a research unit (an academic research group, a synchrotron beamline, an analytical unit in a company, etc.) should not have to buy more than one copy, but that a large institution (university or university department, a synchrotron, a company, etc.) should not just buy one copy and use it for dozens or hundreds of computers. Contact us for detailed information on site licenses.
- *Can I run Datasqueeze from a central server?* See the previous question. There is nothing to prevent you from running Datasqueeze in an X-windows environment from a central server as long as no more than 5 computers have access to it. The program should not be installed on a larger central server to which many users have access.
- *On what types of Windows will Datasqueeze run?* The program should run on all versions of Windows, as long as Java 1.7 or above is installed.
- *Will Datasqueeze run on my Macintosh OS 7/8/9?* No. Mac operating systems older than OS X run an earlier, incompatible version of Java.
- *Will Datasqueeze on my Unix/Linux machine?* It should run fine on any Linux machine that has Java 1.7 or above installed. It might very well run on another flavor of Unix with Java correctly installed, but this has not been tested--contact us if you make it work!! The one caveat is that printing is not currently supported for Linux.
- *Will Datasqueeze run on a computer that is not PC, Macintosh, or Unix?* It should, as long as Java 1.7 or above is installed, but this has not been tested. In any case, some aspects of the program (e.g., the locations of files) are operating-system dependent, so things might not work optimally.
- *I only get part of the graphical interface that I should have--it seems to run off the screen. What is wrong?* Datasqueeze will not work unless the screen resolution is set to at least 1024x768. Assuming that you have a monitor capable of this resolution, you will need to use the Control Panel or System Preferences to increase the resolution.
- *Windows: when I try to launch Datasqueeze I get a panel saying "Unable to launch Java Application or a message saying something about "Could not find main class". What is wrong?* You probably have an older version of Java running—*Datasqueeze* expects at least Java 1.7—or possibly your computer came bundled with JavaSoft, which is Microsoft's version of Java 1.3 and must be removed. Go to the Java site at <http://java.com/en/download/index.jsp> to update to the latest version.
- *Macintosh: when I try to launch Datasqueeze nothing happens, or the icon bounces in the Dock and then quits. What is wrong?* You probably have an older

version of Java running—*Datasqueeze* expects at least Java 1.7. Review the installation instructions

- *Linux: Datasqueeze refuses to start, with an error message saying something about a classpath What is wrong?* You probably have an older version of Java running—*Datasqueeze* expects at least Java 1.7. Review the installation instructions.

File Types

- *What kinds of data file can Datasqueeze read?* Currently supported formats are: “ADSC_Quantum”, “Binary-little-endian”, “Binary-big-endian”, “CrysAlis”, “Bruker-Siemens”, “Bruker-Nonius KCD”, “CBF”, “Ditabis IPC”, “Esperanto”, “ESRF”, “Fuji_IP”, “Gatan DM3”, “MAR_CCD”, “MAR_IP (300)”, “MAR_IP (345)”, “MAR_IP (PCK)”, “Molecular Metrology”, “Nonius DIP”, “Rigaku_Raxis”, “RaxisIV++”, “Roper-Princeton”, “Tiff”, “XDR”, “Unknown.” It is also possible to read in Grayscale graphics images (PNG, GIF, JPG only)--this might be a way of reading in an image that is a Fourier transform of some real-space data, for example produced by a microscope or AFM. (Graphics images are not the preferred way of reading in data from x-ray detectors).
- *Datasqueeze does not read data files produced by my detector. What can I do?* We are continuously adding more formats as the demand arises; contact us at support@datasqueezesoftware.com if there is a particular data format you would like to see added.
- *I saved my data as a GIF (or JPG or PNG) file. I can read it back with Datasqueeze, but the data are quite different. What is wrong?* Graphics files are really intended only as a way of reading in non-xray data, for example produced as a Fourier transform of some optical image. A graphics file is **not** the same as the original data file. A typical data file is 1024x1024 pixels or larger, and has a depth of at least 16 bits (depending on the detector type). Graphics images are generally much smaller (so spatial resolution has been lost) and have much less dynamic range. Additionally, note that *Datasqueeze* expects grayscale images. It converts the image color to an intensity by adding the red, blue, and green components. If you start with a color image, this is almost certainly not how the data were encoded.
- *Can Datasqueeze read in graphics images that are not GIF, JPG, or PNG?* Not right now--expansion of this feature will be driven by customer demand.
- *Datasqueeze returns an error when opening a file that I think is valid. What should I do?* There are several possible origins for this problem:
 1. You may have selected the wrong file format. In the dialog window that opens, there is a drop-down box listing all the data file types that *Datasqueeze* knows about. This should be set to match the type of file you are opening. If you do not know the file type, select “Unknown”.

2. Your file may have been corrupted. This can happen, for example, if you use FTP or a similar protocol to transfer the file from one computer to another. When doing this, you should always use the “binary” option if it exists, rather than “Auto” or “Text.” If you suspect that your file may be corrupted, try transferring it back to the original datataking computer and see if the original software that wrote it can read it.
 3. You may be looking at a graphical image of the file rather than the data file itself. With rare exceptions, a graphics image (jpeg, gif, etc.) has far less information content than the original data file, and in any case is in a different format. Look at the size—a typical x-ray data file is at least 1MB, and often much larger.
 4. If your problem is none of the above, it is possible that this is a data file format not currently recognized by Datasqueeze. Please contact support (support@datasqueezesoftware.com) with as much information as you can provide about the file you are trying to read.
- *My image plate data seem to consist mostly of zeros, even though I know that there is really structure there. What is wrong?* The Fuji image plate data are recorded in logarithmic form. When they are read back in, they are restored to linear form by exponentiation (i.e., $I_{\text{calc}} = A \times 10^{(I_{\text{meas}}/I_0)}$). Depending on the range of the data, it is possible that many or most of your data points are less than 1 (such as 0.01). Try multiplying the data by some large number such as 10000 when you open the file.

Images

- *Image is completely black or completely white.* Try playing with the Max, Min, and Contrast settings in the Image window. If this does not work, go to the Examine window and look at a few regions—make sure you really have read in the data that you think you have.
- *What graphics formats are supported?* The false color image can be saved to disk in the PNG, JPEG, or TIFF formats.
- *What about GIF images?* Sorry, there are copyright problems associated with the GIF format and we chose not to get ourselves in legal hot water.

Calibration/Centering

- *What are x-center and y-center?* In most diffraction applications, the primary beam is blocked by a beam stop. The beam center is the position at which the beam would have hit if it were not blocked. It will be at the center of any Bragg rings. Often this is close to the physical center of the detector, although sometimes it is near one edge of the detector. X-center and Y-center are the values of the pixels corresponding to the center of the diffraction pattern. So for example for a 1024 x 1024 pixel² detector with the beam exactly in the center, X-center and Y-center would both be equal to 512.

- *What is the q-range?* It is the value of $q (= 4 \pi \sin(\theta) / \lambda)$ corresponding to the maximum horizontal range of the detector. This is set once for a particular data set to determine the relationship between the distance in pixels between a feature of interest and the origin and the q-value corresponding to that feature.
- *What is a Bruker standards file and how do I use it?* You are supplied with a file called `agbe.std`, which is a text file containing powder diffraction information on Silver Behenate, a common calibration standard. If you have previously calibrated your detector with AgBe you can use this file with the Calibrate window to easily establish the center, q-range, etc., of your detector. If you have used some other calibration standard, you can easily use any text editor to create a standards file of your own, using `agbe.std` as a model. Note however that this is one place where the number format is not locale-dependent; the d-spacings should all use the American format. That is, a d-spacing that is a little over 45 Angstroms should be written as 45.3 rather than 45,3.
- *Diffraction patterns that should be circular look oval. What is wrong, and what can be done about it?* This can come about in one of several ways. The most common is that your detector is not exactly perpendicular to the incident beam. It is also possible that the x- and y-directions have different gains, or equivalent that the pixels are not exactly square. All of these can be easily corrected by Datasqueeze. The Tilt/Azimuth combination on the Image and Calibrate windows correct for a detector that is tilted an arbitrary amount about an arbitrary axis relative to the central beam.
- *In my images, the beam center is near the bottom of the image. When I set the beam, the entire image is shifted up so that the actual beam center is now at the center of the screen. I can no longer see the upper edge of the image and setting the wedge-shaped region for the plot becomes guess work.* This is actually something that you often want—for example, if you are looking at a small-angle diffraction pattern, you may wish to “zoom” in to the center. The default behavior is to force the image center to be at the beam center. You can get more control by playing with the Zoom command.
- *Datasqueeze Seems to Forget Settings.* In the File window, check the “Retain” box. This will keep beam center, angular range, etc., at their last set value

Plots

- *Plot shows Few or No Data Points:* Check the Plot Limits. Make sure that you actually expect to have data in that range. Make sure also that the Delta value is in a reasonable range—it should be somewhere between 1/10 and 1/100 of the total range of the independent variable.
- *When I go to make a plot it does not graph the full range that I selected.* Check the “Delta” value (the increment corresponding to the spacing between points). If it is larger than the full range of the plot then you will only get 1 or 2 points. If it is much smaller than your plot might actually consist of thousands of points, and

the program cuts off with a maximum number of points. You should typically set the “Delta” so that you have something like 100 points in your plot, but also so that the increment is not smaller than one pixel.

- *When I go to make a plot I get something crazy-looking that oscillates rapidly.* See the previous question. If the “Delta” increment is smaller than one pixel then some points will have data associated with them and some will not.
- *The text file that is produced by saving plots seems to have a lot of blank lines (every other one).* Did you transfer your file from one computer to another? These extra newline characters are sometimes inserted by ftp application (e.g. Fetch, WS_FTP, Filezilla) if it recognizes the file as being “text.” Try telling the ftp that the file is “binary”. This is a slower, but more reliable way, of transferring files. The problem arises different operating systems have different ideas of what constitutes a “new line.” Datasqueeze inserts “\r\n”, which seems to work on all the operating systems we have tried, but is sometimes translated by ftp applications into “\n\n”.
- *Peaks do not Appear at Expected Q-Values or Angles:* This is almost certainly a calibration problem. You need to know enough about your detector to set several parameters. Go to the Calibrate window. First, you need to set Lambda (the wavelength). This is crucial! After that, you have several options:
 1. You can set the Q-range. This is, roughly speaking, the value of Q at one edge of the detector if the beamstop is in the center. You optimize it until peaks appear at the right position. See the manual entry on Calibration for more details.
 2. You can set 2theta-max. This accomplishes the same thing as setting Q-range, for those who are more comfortable using the language of angles than the language of momentum transfer.
 3. You can set both the sample-detector distance and the detector width. (The width should always be the same for a given detector, but the sample-detector distance may vary from one measurement to another).
- *Peaks Appear Broadened or Distorted.* If plotted peaks appear broad or distorted even though the powder diffraction rings in the false color image are sharp, this is almost certainly a calibration problem. There are two obvious things to check:
 1. The beam center (X-center and Y-center in the Calibration window) may not be completely set. Note that the beam center is not in general exactly at the geometrical center of the beam stop. Use the “Circle” option in the Image window to see whether the powder diffraction rings are really centered. Or, make radial plots over limited ranges of Chi and verify that the peaks come at the same positions independent of Chi.
 2. It is also possible that the diffraction rings are not perfectly circular—see the discussion of Oval patterns above. You may need to change the Tilt in the Calibration window.

Least-Squares Fits

1. *Least Squares Fit Failed, Errors Generated.* This can have multiple causes, but the most likely are:

1. Attempting to fit a parameter that actually has no effect on the model.
2. Attempting to fit too many parameters at once
3. Attempting to do a least squares fit to too few data points.

If you suspect #1 or #2, click the Revert button, deselect one of the parameters that you fit, and try again. If the number of parameters is greater than or equal to the number of points, you will either need to redo the plot so as to increase the number of points, or decrease the number of fitting parameters.

Run-time errors

- *Datasqueeze unexpectedly crashed. What went wrong?* If you generate a reproducible error please contact us with as much information as you can supply and we will try to fix the problem. (Most such bugs have already been eliminated).
- *I got an error message that said that Java is out of memory. What went wrong?* If you get an out of memory error, you may be trying to read in a larger file than Datasqueeze can hold in memory.

Known Bugs

- **Dialog windows get hidden:** If you open a dialog window (for example, to open a data file or save a graphics image), the main program waits until this window has completed its operations and been closed. This can cause a problem if you temporarily switch to a different application and then switch back--you may find that the window is now hidden behind the main application but the main interface window does not respond. If this happens, first try dragging the offending window to one side--the dialog box may reappear from underneath it. If this does not work, your only option is to forcibly kill the Datasqueeze application (for example, with Force Quit on Mac, or kill -9 on Linux, or Alt+F4 on Windows). To avoid this situation, before switching to a different application if a Dialog is open try to drag it so that at least part of the Dialog overlaps with the main desktop.
- **Funny Plots:** Again, not really a bug. If you get an extremely noisy plot, or if only some of the data are plotted, check the "Delta" box in the plot parameters. For example, suppose you are doing a Q/Chi plot, and the full range in your detector is 0 to 0.2 inverse angstroms. You had Qmin set to 0.01, Qmax set to 0.18, and Delta set to 0.0001. This would imply 1700 data points, which is probably more data points than there are pixels in that range. Similarly, if you set Delta to 0.3 there will be only two points in the plot. Typically you want to set Delta to something like the greater of (a) the total range divided by 100 or (b) the estimated number of pixels along a line connecting the start and end points of the independent variable.
- **Macintosh—Print problems:** On OS-X, Datasqueeze responds badly if something goes wrong when printing. In particular, when printing multiple pages, it is important to make sure that one page has completely finished printing before initiating another print job. Not doing so seems to put the program into a funny mode from which it is may be necessarily to forcibly kill the application using Force Quit. We are working to locate the origin of this bug.
- **Zoom/Examine:** The Examine panel behaves confusingly if the zoom is changed while the data are being examined. For consistency, choose the desired Zoom value in the Appearance Panel before doing the pixel-by-pixel examination.
- **"Corrupted" File** Sometimes the file opening panel will claim that a file is corrupted. There are actually two possibilities here. One is that you really have a bad data file. The other (more likely) is that you are trying to use the wrong file format. Remember to check the "file format" drop-down menu next to the file navigation widow and make sure that the format is appropriate for your data file type. If you are not sure, try "unknown"--this is slow but usually gets the right answer.

Feedback

Please report any bugs, anomalies, or requests for features to support@datasqueezesoftware.com. If you are reporting a bug, provide as much information as possible on exactly what you were doing when the bug surfaced, what platform you were using (e.g., PC running Windows ME), and the extent to which it is reproducible.

Note on Copyrights

Bruker is a trademark of Bruker AXS.

ADSC is a trademark of Area Detector Systems Corporation.

MAR is a trademark of X-ray Research GmbH.

Fuji is a trademark of Fuji Photographic Film Inc.

Molecular Metrology, Rigaku and **R-Axis** are trademarks of Rigaku/MSK.

Gatan is a trademark of Gatan, Inc.

Ditabis is a trademark of Ditabis Digital Biomedical Imaging Systems AG.

Java is a trademark of Oracle.

About the Author

Datasqueeze was written by Paul A. Heiney, a professor of physics at the University of Pennsylvania. See his web page at <http://dept.physics.upenn.edu/~heiney/index.html>. He is an author of more than 150 refereed publications, most having to do with x-ray scattering in some form. He is a member of The American Chemical Society, The American Crystallographic Association, and other scientific societies, and is a Fellow of The American Physical Society.

Along with Prof. Karen Winey, Heiney is responsible for the Multiple-Angle X-ray Scattering Facility (<http://www.lrsm.upenn.edu/lrsm/facMAXS.html>) at the Laboratory for Research on the Structure of Matter. This facility is open on a paid basis to scientists outside Penn.

Datasqueeze was originally written for analysis of x-ray data at the University of Pennsylvania, and all versions of the program have been extensively used at Penn. Datasqueeze Software was founded in 2002 to disseminate the program to a wider audience.

Index

- 1D Data, Import, 56
- 2DFFT, 137
- 2-theta, 40
- 2theta-max, 37
- 3D Display, 21
- 3D Image, Save, 57
- Access Key, 64
- Add, 15, 16, 68
- Add Frame, 21
- ADDMOVIEFRAME, 21, 110
- ADSC, 148
- ADSC_Quantum, 124, 125, 142
- Advanced Features, 46
- Advanced Plot Features, 56
- agbe.std, 30, 35
- Aggregate, Fractal, 103
- Angle, azimuthal, 40, 43, 44, 47, 54, 61, 76, 79, 147
- Angle, scattering, 40
- Angles, Incorrect, 145
- Antialias, 46
- Append, 16
- APPENDTOFILE, 110
- APPENDTOFILE, 16
- Applications, 12, 13, 68
- Apply Button, 51, 53, 80
- APPLYFIT, 111
- APPLYMASK, 111
- ascii 2-column profile, 58
- ascii profile, 58
- Ascii, Save As, 57
- ASCII2COLUMN, 115
- ASCIIPROFILE, 115
- Autocalc, 59, 63
- Autocalibration, 28
- Average, 42, 43, 76, 78, 87
- Axes, 43, 78
- AXISRANGE, 111
- AZIMUTH, 111, 144
- azimuth, detector, 31
- Batch, 60, 61, 62, 66, 109, 110, 137
- Batch Help, 66
- Beam (Q=0) Center, 19
- beam center, 30, 32
- BEAMCEN, 112

Bessel, 105, 134
 Bessel Function, 104
 Bessel Function Squared, 105, 134
 Binary-big-endian, 124, 125, 142
 Binary-little-endian, 124, 125, 142
 Blowup, 54
 Boolean variables, 138
 Bragg rings, for calibration, 29
 Browse, 15, 16, 68, 74
 Bruker, 6, 10, 12, 68, 144, 148
 Bruker calibration file, 30, 35
 Bruker-Nonius KCD, 124, 125, 142
 Bruker-Siemens, 124, 142
Bugs, 147
 CALCSTATISTICS, 112, 130
 Calibrate Panel, 28, 69
 Calibration file, 6, 10, 12
 Calibration Parameter Source, 15
 Calibration Parameters, Manual, 36
 Calibration Parameters, Save/Retrieve, 35
 Calibration, Verify, 35
 Camera Length, 37
 Cancel Button, 16, 52
 Cancel, Fit, 52
 CBF, 142
 CEN, 109
 Center, Data, 19
 Center, User Defined, 19
 Centering, 36, 143
 Centering, Image, 19
 Check for Updates, 67
 Chi, 40, 43, 44, 47, 54, 61, 76, 79, 147
 Chi Offset, 38
 CHIOFFSET, 112
 CHIOFFSET, 39
 Chi-Squared, 83
 Coated Cylinder, 102
 Coated Cylinder, 101, 134
 Coil, Gaussian, 102, 103, 134
 Color Scale, 17
 Color Scheme, 17
 Command Log, 14
 COMMENT, 112
 completely black, 143
 completely white, 143
 Condense, 18, 23, 24, 113

Construct Button, 50

CONSTRUCTMODEL, 50, 113, 128

Contour, 18

Contour, levels, 18

CONTOURSCALE, 113

CONTRAST, 113

Contrast, Color, 17

control-click, 13

Copy, 60

Copy Fourier Transform Image, 60

Core-Shell, 96, 97, 134

correlation coefficient, 52, 82, 85, 89

Correlations Button, 52

corrupted, file, 147

cpi, 58, 115

CrysAlis, 124, 142

csv, 58, 115

Cursor, Show, 46

Cursors, Fit, 80

Cylinder, 100, 101, 134

Cylinder, Coated, 101, 102

Data Center, 19

Data, Export, 58

Data, Plot, 58, 62, 77

Data, Real, 44, 74

dbws, 58, 115

Defaults, Restore, 22

Delta, 41, 76, 144, 145, 147

Delta, Plot, 42

Detector 2Theta, 29, 38, 40

Detector Azimuth, 29, 38, 40

Detector Diameter, 38

detector orientation, 31

Detector Tilt, 29, 37, 39

Detector, diameter, 114

Detector, Distance, 119

Detector, pixel size, 121

DETECTORAZIMUTH, 113

DETECTORTWOTheta, 114

De-Zing, 23

DEZING, 114

Dialog windows, 147

DIAMETER, 114

Disable Automatic Image Rescale, 22

Disable Center Change, 36

Disable Image Display, 22

Disk, 99
 Dispersion, 63
 Display Fit Cursors, 53
 Display,3D, 21
 Ditabis IPC, 124, 125, 142
 Divide by $A + B \cos^2(2 \theta)$, 47
 DM3, 124, 125, 142
 DOFIT, 114
 Double variables, 138
 Download, 5
 Edit Masks, 64
 Edit Menu, 60
 Ellipse, 40, 41
 Ellipsoid, 97, 98, 134
 Enable Center Change, 36
 Enter Instrument Parameters by Hand, 35, 36
 Error Bars, 46, 53, 59, 82, 84, 85, 89, 114, 135
 Errors, Statistical, 87
 Esperanto, 124, 125, 142
 ESRF, 124, 125, 142
 Examine Panel, 54, 75
 Exclude Points In Column, 65
 Exclude Points In One Pixel, 65
 Exclude Points In Row, 65
 Exclude Points Inside Box, 65
 Exclude Points Inside Circle, 65
 Exclude Points Outside Circle, 65
 Exit, 60, 147
 Export, 58
 EXPORTPLOT, 115
 EXPORTPLOT, 58
 False Color Image, Save, 57
 False Minima, 85
 FAQ, 141
 Fast Fourier Transform, 25
 Few or No Data Points, 144
 FFT, 25
 File does not open, 142
 File Format, ADSC, 148
 File Format, ADSC_Quantum, 124, 125, 142
 File Format, Binary-big-endian, 124, 125, 142
 File Format, Binary-little-endian, 124, 125, 142
 File Format, Bruker, 6, 10, 12, 68, 144, 148

File Format, Bruker-Nonius KCD, 124, 125, 142

File Format, Bruker-Siemens, 124, 142

File Format, CBF, 142

File Format, CrysAlis, 142

File Format, Databis IPC, 124, 125, 142

File Format, Esperanto, 124, 125, 142

File Format, ESRF, 124, 125, 142

File Format, Fuji, 143, 148

File Format, Fuji_IP, 124, 125, 142

File Format, Gatan DM3, 124, 125, 142

File Format, GIF Image, 142, 143

File Format, GrayScale Image, 124, 125

File Format, MAR, 148

File Format, MAR_CCD, 124, 125, 142

File Format, MAR_IP, 124, 125, 142

File Format, Molecular Metrology, 124, 125, 142, 148

File Format, Nonius DIP, 124, 125, 142

File Format, PNG Image, 57, 62, 128, 130, 142, 143

File Format, RaxisIV++”, 124, 125, 142

File Format, Rigaku, 148

File Format, Rigaku_Raxis, 124, 125, 142

File Format, Roper-Princeton, 124, 125, 142

File Format, SBIG, 124, 125

File Format, TIFF, 57, 62, 128, 130, 142, 143

File Format, Unknown, 124, 125, 142

File Info, 63

File Menu, 40, 54, 55, 75, 77

File Name on Image, 20

File Panel, 15, 44, 55

File, Calibration, 6, 10, 12

File, Running from, 60, 61, 66, 109, 110, 137

Files to Read In, Number, 15

FILESWAP, 115

Fit, 50, 51, 53, 60, 61, 62, 66, 80, 81, 86

Fit Button, 51, 53

Fit Cursors, 80

Fit Cursors, Display, 53

Fit environment, restore, 51

Fit environment, save, 50

Fit Failed, 145

Fit Help, 50, 66

Fit Message Area, 51

Fit Panel, 50, 51, 53, 81

Fit Parameter Panel, 50, 51, 53, 80
 Fit, Cancel, 52
 Fit, Least-Squares, 50, 51, 53, 54, 55, 58, 59, 61, 62, 66, 79, 83, 85, 87, 89, 114
 Fitting Parameter, 51
 Fitting Parameter, Apply, 51, 53
 Fitting Parameters, 59, 60, 62
 Fitting Parameters, Correlations, 52
 FITVSQ, 115
 FITVSQ, 50
 Flip Horizontal, 24
 Flip Vertical, 24
 FLIPFIVERTICAL, 115
 FLIPHORIZONTAL, 115
 Font Size, 49
 Font, Plot, 134
 Font, plot, 49
 FORCEGRID, 116
 Forget Settings, 144
 Format Info, 63
 Format, File, 63
 Fourier Transform, 25
 Fourier Transform Image, Copy, 60
 Fractal Aggregate, 103, 104, 134
 Frequently Asked Questions, 141
 Fuji, 143, 148
 Fuji_IP, 124, 125, 142
 Function Selection, 50
 Functions, Number of, 50, 79
 Funny Plots, 147
 Gatan DM3, 124, 125, 142
 Gaussian, 79, 86, 91, 134
 Gaussian Coil, 102, 103, 134
 GIF, 142, 143
 GrayScale Image, 124, 125
 Grid Intervals, Fix, 20
 Grid Intervals, Set, 20
 Grid, Plot, 49
 Grid, Show, 20
 GRIDINCREMENTS, 116
 gsas, 58, 115
 Guinier, 40, 44
 Guinier Plot, 44
 Heiney, Paul, 148
 Help, 50, 66, 79
 Help Menu, 66

Help, Batch, 66
 Help, Fit, 50, 66
 HORIZSYM, 116
 Hue, 17
 Im #, 15
 Image Centering, 19
 Image Display Rescale, Disable, 22
 Image Display, Disable, 22
 Image Panel, 17, 54, 75
 Image Window, 14, 51, 77
 Image Zoom, 19
 Image, False Color, 14, 51, 57, 60, 75, 77
 Image, Plot, 58, 60
 IMAGEAUTORECALCENABLE, 117
 IMAGEDISPLAYENABLE, 117
 Images, 143
 Import 1D Data, 56
 Import Masks, 65
 IMPORT1D, 118
 Independent parameter, 83, 87
 Index, Image, 15
 INITIALIZEMOVIE, 21, 51, 117
 Installation, 5, 6, 8, 11, 12, 141
 Integrate, 42, 43, 76, 77, 87
 Integrate/Average, 43
 INVERTSYM, 118
 Java, 6
 Java, locating, 7
 Kinning-Thomas, 106, 107, 134
 Lambda, 29, 37, 39, 40, 118
 Least-squares fit, 50, 51, 53, 54, 55, 58, 59, 61, 62, 66, 79, 83, 85, 87, 89, 114
 LENGTH, 118
 Levels, Contour, 18
 Limits, Plot, 41, 76
 Line, 41, 48, 80, 133
 line plot, 23, 53, 54, 59, 79, 130
 Line Width, 47
 Line width, 132
 Linear, 17, 18, 55, 75, 78
 LINEARSCALE, 119
 Lines, Diffraction, Save, 58
 Lines, Powder, 45
 Linux, 6, 11, 12, 13, 57, 58, 59, 68, 75, 123, 141, 142, 147
 Log, 14, 61, 78, 110
 Logarithmic, 17, 18, 75

log-normal, 95, 96, 100
 LOGSCALE, 119
 Lorentz factor, 43, 47, 120
 Lorentzian, 90, 92, 134
 Lorentzian-Squared, 92, 93, 134
 Macintosh, 11, 13, 55, 57, 58, 59, 60, 61, 63, 66, 68, 109, 141, 147
 Make/Update Plot, 42
 MAKE3DIMAGE, 119
 Manipulate Panel, 23
 MAR, 148
 MAR_CCD, 124, 125, 142
 MAR_IP, 124, 125, 142
 Marker Size, 48
 Marker Type, 48
 Marquard algorithm, 87
 MASK, 119
 Mask Menu, 64
 Maximum Q, 37
 MAXVALUE, 120
Memory, 146
 Menu, Help, 66
 Menu, Tools, 65
 Menu, View, 63
 Minima, False, 85
 MINVALUE, 120
 Molecular Metrology, 124, 125, 142, 148
 Momentum transfer, 40, 41, 43, 44, 47, 54, 61, 77, 79, 109, 122, 130, 147
 Movie, 61
 Movie Controls, 20
 Movie, New, 21
 Movie, Save, 21, 59
 MPEB Box, 53
 Mult, 15, 16, 68
 Multi-Parameter Error Bars, 51, 53
 Multiply by $\sin(2\theta)$, 47
 Multiply by $\sin(\theta)$, 47
 Multiply by x^n , 47
 MULTIPLYSINTHETA, 120
 MULTIPLYSINTTHETA, 120
 NCONTOUR, 120
 New Instrument Parameters, 68
 NMASK, 121
 NMODELS, 50, 121
 Nonius DIP, 124, 125, 142
 Number of Files to Read In, 15

Number of Functions, 50, 79
 NUMSTATBINS, 121
 one-parameter error bars, 82
 Open, 15, 16, 44, 55, 69, 74
 Open Recent Files, 55
 Operations, Order of, 16
 OSX, 147
Out of memory, 146
 Overlay, Save, 20
 Oxford Instruments, 124, 142
 Page Setup, 59
 Parameter entry, 51
 Parameter, independent, 83, 87
 Parameters, Strongly Coupled, 85
 Percus-Yevick, 106, 107, 108, 134
 Pixel Size, 38
 PIXELSIZE, 121
 Play, 21
 PLAYMOVIE, 21
 PLAYMOVIE, 122
 Plot, 14, 40, 41, 43, 44, 54, 55, 58, 59, 60, 61, 62, 63, 76, 77, 78, 79, 80, 87, 109, 111, 112, 122, 123
 Plot Delta, 42
 Plot Features, Advanced, 46, 56
 Plot Font, 49
 Plot Grid, 49
 Plot Image, Save, 58, 62
 Plot Limits, 41, 76
 Plot, Contour, 18
 Plot, Guinier, 40, 44
 Plot, Increment, 41, 76, 144, 145, 147
 Plot, Line, 23, 41, 48, 51, 53, 54, 59, 77, 79, 80, 130, 133
 Plot, Pole Figure, 44
 Plot, Porod, 44
 Plot, Powder, 43, 44
 Plot, Preset, 43, 44
 plv, 58, 115
 PNG, 57, 62, 128, 130, 142, 143
 Polarization correction, 47, 131
 Pole Figure Plot, 44
 Polynomial, 79, 89, 134
 Porod, 44
 Porod Plot, 44
 Powder Diffraction Lines, 45
 Powder Diffraction Lines, Save, 58
 Powder Plot, 43, 44

Powder Rings, Show, 22, 46
 Power Law, 93
 Preferences, 59, 60
 Presets, 43, 44
 Princeton, 124, 125, 142
 Print, 59, 62, 78, 123, 147
 Process Multiple Files, 61, 62
 Q, 40, 43, 44, 47, 54, 61, 76, 77, 79, 109, 122, 130, 147
 Q - S - D - Theta Calculator, 65
 Q as Fit Variable, 50, 80
 Q-range, 29, 37, 39, 40
 Qrange, 43, 44, 123
 Quit, 60, 147
 Q-Values, Incorrect, 145
 Qx, 40, 41, 54
 Qy, 40, 41, 54
 Radius of gyration, 44, 94, 102
 Radius-Gyration, 134
 RaxisIV++”, 124, 125, 142
 Rayleigh, 95, 96, 105, 106, 134
 READFILE, 16, 55, 109, 124
 READIN, 124
 RECALCIMAGE, 109, 110, 125
 Recalculate, 63
 Recent Files, 55
 Requirements, 5, 6, 11
 Reset button, 42, 53
 Resolution, Saved Image, 22
 Restore, 60
 Restore Defaults, 22
 Restore Environment, 51
 RESTOREDEFAULTIMAGEVALUES, 125
 RESTOREFITENVIRONMENT, 126
 RESTOREFITENVIRONMENT, 50
 Retrieve Parameters button, 35, 36
 RETRIEVEINSTRUMENTPARAMETERS, 126
 RETRIEVEINSTRUMENTPARAMETERS, 34, 36
 Revert Button, 53
 Rigaku, 148
 Rigaku_Raxis, 124, 125, 142
 right click, 13
 Rod, 98
 Roper-Princeton, 124, 125, 142
 Rotate, 24, 126
 Sample, 68

Sample-Detector Distance, 37
 Save, 57, 58, 59, 60, 62, 75, 77
 Save 2D Data as Tiff, 58
 Save 3D Image, 57
 Save as Ascii, 57
 Save Calculated Diffraction Lines, 58
 Save Environment, 50
 Save Masks, 65
 Save Movie, 21, 59
 Save or Retrieve Instrument Parameters, 35
 Save Overlay, 20
 Save Parameters button, 36
 Save Plot Image, 62
 SAVE3DIMAGE, 128
 Saved Image Resolution, 22
 SAVEDATA, 126
 SAVEDATAASTIFF, 127
 SAVEFITENVIRONMENT, 127
 SAVEFITENVIRONMENT, 50
 SAVEFITPARAMETERS, 128
 SAVEIMAGE, 109, 128
 SAVEINSTRUMENTPARAMETERS, 129
 SAVEMOVIE, 129
 SAVEOVERLAY, 129
 SAVEPLOT, 58, 109, 129
 SAVEPLOTIMAGE, 129
 SAVESTATISTICS, 130
 SAXS, 28, 37, 39
 SAXS Mode, 39
 SBIG, 124, 125
 Scaling, Axis, 49
 Script, 60, 61, 62, 66, 109, 110, 137
 SELECTREGION, 130
 Set Intervals, 20
 SETANTIALIAS, 130
 SETCOLOR, 131
 SETCONTOUR, 131
 SETCOSINEAB, 131
 SETFITPARAMETER, 51, 132
 SETGRID, 132
 SETINTENSITYEXPONENT, 131
 SETLINEWIDTH, 132
 SETMARKERINDEX, 133
 SETMARKERTYPE, 133
 SETMODEL, 50, 134

SETPLOTGRID, 134

SETPLOTGRIDCOLOR, 134

SETSTATBIN, 135

Settings, 59, 60

Show Bragg Ring Positions from File, 35

Show Cursor, 46

Show Grid, 20

Show Powder Rings, 22, 46

Show Text, 46

Show/Hide Masks, 64

SHOWEB, 135

SHOWTEXT, 135

Silver Beheneate, 144

Sine, 94, 95, 134

Size, Font, 49

Small-Angle, 28, 39

Smooth, 23, 24, 135

Solaris, 7, 12

Startup, 13, 79

Statistical Errors, 87

Statistics, 58, 62, 63, 78

Stop, 21

String variables, 137

Submodels, 53

Submodels Box, 53

Sum, 42, 43, 77, 87

Summed Counts, 43

Symmetrize Horizontally, 24

Symmetrize Vertically, 24

Symmetrize, Inversion, 25

Text, Show, 46

ThinDisk, 100, 134

ThinRod, 99, 134

TIFF, 57, 62, 128, 130, 142, 143

Tiff, Save As, 58

TILT, 135

Tilt Azimuth, 29, 37, 39

Tilt, Detector, 111, 135, 144

Tools Menu, 65

TWOTHETARANGE, 135

Type, Marker, 48

University of Pennsylvania, 148

Unix, 7, 12, 59, 123, 141

Unknown, 124, 125, 142

Updates, Check, 67

Use Q as Fit Variable, 50, 80

User Defined Center, 19

Value, Color, 17

Variable, Dependent, 40, 43, 44

Variable, Independent, 40, 42, 43, 47, 55, 86, 87, 122, 130, 147

Verify Calibration, 35

Version History, 139

Version, Trial, 5, 10, 11, 13, 141

VERTSYM, 136

View Menu, 63

Voigt, 92, 134

wavelength, 29, 37, 118

WAXS, 29, 40

WAXS Mode, 39

WAXSMODE, 40, 136

Wide Angle Mode, 29, 38, 40

Windows, 5, 13, 55, 57, 58, 59, 60, 68, 75, 109, 141, 147, 148

Windows Menu, 67

WRITEFILENAME, 136

X and Y Axis Options, 49

x-center, 143

X-windows, 10, 11, 141

Yarusso-Cooper, 105, 106, 134

y-center, 143

Zoom, 18, 37, 78, 136, 147

Zoom In, 64

Zoom Out, 64

ZOOMCEN, 137