CCD\_to\_qspace: MATLAB package (Kiyo Akabori, 10/10/2014)

This document describes analysis of near grazing incidence wide angle X-ray scattering (nGIWAXS) using MATLAB. Open an example script file named “waxs.m” under “scripts” directory. It is a script that was used to analyze ripple phase nGIWAXS data (partially modified for this tutorial).

For any analysis with MATLAB, I suggest you to prepare a similar script instead of doing everything through the MATLAB command line. Script files allow you to reproduce exactly what you did. For debugging, the command line might be more useful. Below, I outline WAXS analysis. Comments are also included in waxs.m.

1. A script can be stored anywhere as long as you use the absolute paths to the MATLAB functions and data.
2. Find out the absolute path to your “CCD\_to\_qspace” directory. Some examples are
   1. (Linux) /home/kiyo/WinE/MATLAB\_UserFunctions/Functions/CCD\_to\_qspace
   2. (Windows) C:\Documents and Settings\Owner\My

Documents\MATLAB\_UserFunctions\Functions\CCD\_to\_qspace

1. Find out the absolute path to your data file
2. The following two lines tell MATLAB where files are located:

>> addpath(genpath('path\_to\_your\_CCD\_to\_qspace\_directory’));

>> addpath('path\_to\_your\_data\_directory');

1. Set up three global variables: wavelength, pixelSize, and sDist, where
   1. wavelength: X-ray wavelength
   2. pixelSize: pixel size, usually 0.07113 mm/pixel
   3. sDist: sample-to-detector distance
2. Load CHESS data and subtract pedestal and background scattering as usual
3. Invert the background subtracted image by entering

>> image = flipud(image)

1. Find the beam position (x, z). Note that image orientation is similar to tview because of the flipud command above. Save the values as beamX and beamZ variables.
2. To convert the CCD image into q-space, add the following command:

>> q\_image = transform\_ccd2q(image, [0.6 1.8], [-0.6 1], 0.0024, 0.0024, -45, beamX, beamZ);

[0.6 1.8] is the qr range that you want to be converted. The range can be larger than the probed q range, but you will simply have black pixels for the q-space that was not probed in the experiment.

[-0.6 1] is the qz range. For nGIWAXS data, you want the lower range to be 0 or greater. For tWAXS data, you probably want both above and below the equator. Here, tWAXS data is assumed.

The first 0.0024 is the step size in qr. This value should be approximately Δq ≈ 4π Δθ / λ ≈ 4π pixelSize / (λ sDist). It’s basically the angular resolution of a pixel in inverse Angstom.

The second 0.0024 is the step size in qz.

-45 is the angle of incidence in degrees. If the beam hits the substrate first, and then hits the sample, this value should be negative (tWAXS). For nGIWAXS, this value is somewhere between 0.2 and 0.5.

beamX and beamZ are the horizontal and vertical beam position.

1. To create a 2D image with proper qr and qz axes, enter the following commands:

>> fig1 = figure

>> qshow(q\_image, [grayscale]);

Here, we save the state of a plot in the fig1 variable, which can be used to save an image later. qshow is an analog of Mill’s show command.

1. To add axis labels, enter

>>

1. To save a 2D image, enter the following command:

>> saveas(fig1, “filename.pdf”);

1. To make a qz swath and save as an ASCII file, run:

>> figure

>> [qz, Int] = qzplot\_q(q\_image, [qr\_low qr\_high]);

>> dlmwrite('filename.dat', [qz Int]);

1. To make a qr swath and save as an ASCII file, run:

>> figure

>> [qr, Int] = qrplot\_q(q\_image, [qz\_low qz\_high]);

>> dlmwrite('filename.dat', [qr Int]);

1. To make a q swath (so called sector plot),

>> figure

>> [q, I] = sector\_q(q\_image, [q\_low q\_high], [phi\_low phi\_high]);

>> dlmwrite('filename.dat', [q, I]);

1. To make a phi swath,

>> figure

>> [phi, I] = integrate\_annulus\_q(q\_image, q\_range, phi\_range, bin\_size, N\_point, q\_delta);

>> dlmwrite(‘filename.dat’, [phi, I]);