Guide to taking force curves using GXSM.

William Elferink

April 17, 2017

mail: william.elferink@gmail.com

Why use GXSM for force curves?

The original set-up for acquiring force curves with the Colour AFM as described in Colour AFM Manual V12 uses a function generator and a oscilloscope to acquire force curves. Though this is a flexible and easy to use method for acquiring the data, it is only capable of obtaining force curves one by one, which have to be saved, transferred to a computer and processed individually. Doing this saving and transporting of every force curve manually can become an tedious process when a larger number of force curves are required or desired.

The only thing that is needed for force curve acquisition is a triangle wave for the Z-piezo with simultaneous frequency shift detection. Instead of using a function generator for generating this triangle wave we can also use a GXSM scan, because the X-piezo output wave also happens to be a triangle wave and the simultaneous acquisition of data is obviously built in for imaging purposes!

This guide explains how to obtain force curves with GXSM, which can obtain thousands of force curves (as many as you'd like) requested in a single measurement. The converted version of the files generated by GXSM (as explained later) can be put into the Python program I provided to analyse them and save all the individual force curves in their own graphs, with the right axis formatting (assuming your GXSM settings are correct). From these multiple force curves you can even visualize the vertical drift rate!

An idea for determining Adamantane-trithiol's effect on the force curve depth.

One idea for reducing the van der Waals contribution to the force curve is to functionalise the tip of the AFM with a Adamantane-trithiol molecule. Removing this background force would make the obtained force curve more accurately represent the Morse potential. Should the necessary lateral accuracy of obtaining force curves on these specific molecule sites turn out to be hard to achieve, it might be an idea to execute this measurement with a statistical approach.

If we have two samples, one sample of Gold(111) and one sample of Gold(111) with randomly deposited Adamantane-trithiol molecules, we could see the effects of these molecules by comparing the force curves made (with the same

tip) on both these samples. If we make e.g. 5000 force curves on each sample, while letting the tip walk randomly in the XY-plane (by timing a XY voltage change with the force curve acquisition), there should be a difference on average between the depth (E_b), width (λ) and position of the minimum (Z_0) of the force curve. By extracting these data values by writing an extension to the Python program, we can form a histogram of the values for both samples. In this histogram one should see that the E_b values of the sample with the Adamantane-trithiol molecules are lower than that of the pure Gold(111) sample.
A quick note on the random XY-voltage that needs to be applied: the timing can be made easier by executing the experiment through the PyRemote plug-in of GXSM. This Python-based measurement automation plug-in can be used to, for example, pause the measurement after every scan-line for a few seconds or whatever is desired. For reference for this plug-in see the GXSM manual found on their website (old, slightly outdated version).

Guide to acquiring force curves

To get a force curve through GXSM, we first have to find the right amplitude and offset for the signal in Z. So to start the process we first use the traditional method of force curve acquisition:

1. Autoapproach as usual.
2. Connect the function generator's output to the "Add Z" input on ZeeKon.
3. On the function generator, set the wave to a triangle wave and play with the amplitude and offset until the desired force curve is visible on the oscilloscope.
    - For a block diagram of the original force curve acquisition set-up, see the Colour AFM Manual V12 written by Denis Damiron.

Once you have obtained a force curve the traditional way, we need to recalculate the voltage parameters from the function generator to their GXSM X range equivalent. This is done by converting the function generator's amplitude into the X Range value in the GXSM scan window, by multiplying it with the Angstrom/Volt for the X-piezo as found in the GXSM settings. For example, with the current settings (April 17, 2017), an amplitude of 3.4V peak-to-peak would result in Range X = 510 Angstrom when X-Piezo-AV is set to 150 A/V in GXSM (make sure the VX gain in GXSM is set to 1, or the SoftDB SPM Controller will output different voltages).

1. Disconnect the X scan, X offset, Y scan and Y offset cables from the SoftDB SPM Controller.
2. Connect the function generator and the X scan output of the SoftDB SPM Controller to an analogue adder, and connect the adder's output to the "Add-Z" input of ZeeKon.
3. Make sure the ΔF signal from the Quad Voltage buffer is connected to the DFdc input of the SoftDB SPM Controller. (Check if the InstSPM →

Instrument/dHertz2Volt setting is on the same setting as Highdi (100Hz on April 17, 2017)).

4. Input the calculated number for Range X into its field.
5. Set the "Points" value for X to the desired amount of points that will be taken in a force curve. Points in Y will be adjusted to the same number automatically.
6. Set the Scan speed to the desired value (calculate time from: Time per force curve = $(2 * XRange)/Scanspeed$, where the two comes from having to move right and left)
7. In the channel settings, set the first and second channel to DFdc, and in the third column "Dir", select $\rightarrow$ for Channel 1 and $\leftarrow$ for Channel 2.
8. If you also want to capture the current, connect it to the ITunnel channel and repeat the step above for channels 3 and 4.
9. **For the next step, keep a close eye on the values as GXSM gets buggy and will reset your values often.**
10. In the second column for the points, adjust the value for the amount of force curves that you want to take. After entering the number, click in the Range Y text field. **Don't click *any* other text field or button or the value for Points Y will reset to equal Points X**.
11. If the value in Points Y has not changed, click the start scan button.
12. Wait until the scan has finished and save each channel with an easy to identify name (include scan direction in the filename, you need to be able to identify which was which).

Now we have our force curves! There is just one problem, the *.nc file format outputted by GXSM cannot easily be read by Python. To make my life easier in Python coding, I used a conversion tool to convert the *.nc file into a *.txt file. This plain text file allows for easier juggling with the data. To convert the data:

1. Open a terminal.
2. Check if the conversion tool is installed by typing: sudo apt-get update
3. After the process has completed: sudo apt-get install netcdf-bin
4. If it was not installed, it will be now.
5. Using the navigation command "cd" navigate to the directory in which the *.nc file is present. Click here for a guide on folder navigation.
6. Now type: ncdump filename1.nc > filename1.txt
    - Do this for each of the files.

These files should be transferred to a PC with a Python installation (Anaconda installation because of the dependencies needed) where they can be processed by the Python program. To run the script:

1. Install Anaconda Take the 64 bit installer Python 3.5 version (32-bit if your Windows is 32 bit, but it most likely is not).
2. Once installed, a program called Spyder (a Python editor) came with it. Open this application.
3. Obtain my bundled Python script .zip from one of your colleagues's mail.
4. Unzip the .zip file, and put the contents in a place of your choosing.
5. In Spyder, open the ForceCurveAnalyzer.py file as found in the forcecurve-analyzer folder.
6. Press F5, to run the script. From here it should be straightforward.