Comparison of the function elementPredictor2000 and lattice_analyzer2011

The existing function elementPredictor2000 is a function that, given the distances from the outgoing hole to the measured values and given the wavelength, calculates the lattice constant for X-ray powder diffraction. It is based on an algorithm, that iterates over all possible lattice constants and all given data points. Then it calculates the theoretical values that would be measured and compares them for all the different lattice constants to find the best fit. It is working well, however there are a couple of things it does not include.

First of all the film undergoes shrinkage while drying. In the experiments I have done this was a small effect (less than 5% of original length), nevertheless it causes a small error. Secondly the program uses only one wavelength, in my experiment a copper target with a nickel filter was used. The nickel filter blocks the Kβ lines and lets the Kα lines pass, which means there are two slightly different wavelengths for the Kα1 and Kα2 lines. They are not always but sometimes distinguishable. A third point is that the Nelson-Riley-plot gives very precise values for the lattice constant of atoms. This function only minimizes the squared errors, although it might seem logic to minimize the reduced chi^2 of the Nelson-Riley. As the fourth point I would like to mention that I sometimes wanted to watch the fitted data and get some more details about it. This function has a minimum data input and output, which is good, but as a mentioned sometimes it is nice to be able to get more. The last and fifth point is less important, it is just that I find it easier to write the data into an Excel file and import it automatically from there rather than having to import it manually into Matlab.

The program lattice_analyzer2011, which I wrote, solves most of these problems. It includes the shrinkage, and thus solve the first mentioned point, by assuming the full circumference is the measurement for the outgoing hole minus the measurement for the incoming hole divided by two. As the angles are calculated relative to this value the shrinkage is thereby covered. The second point is solved by not only iterating over all different miler indexes, but also iterating over all the different wavelengths, minimizing them, too. To solve the third point my function first dose a rough fit minimizing the squared errors just as the other function, this returns the ten best lattice constants. Then for each of them a chi2 fit is down and the value with the minimal reduced chi2 is taken. In the high resolution fit even the first is done minimizing the reduced chi2 instead of the squared errors. It iterates over 80 different values in a distance of 5 pm each. The fourth point is solved in connection with the fifth point, it is more difficult to present a lot of different output values so, that you can easily read them. Therefore I decided to make the input and the output of the function an Excel sheet. In addition to that the function returns a hole bunch of data to Matlab, but for sure it will take a closer look to figure out which is which there, so this is really only if one wants do further analyze the data in Matlab.

Still there are a bunch of things that can be further improved in my function. It takes the wavelength into account, but as there is errors in the data and the wavelength is optimized before doing the Nelson-Riley fit, the program sometimes guesses the wrong wavelength although you can clearly see which is right on the film. An easy way to fix this might be to have the possibility to fix the wavelength, when giving the data in the Excel sheet. Also in the current function just a bunch of lattice constants are given to the program. I found online the Crystallography Open Database (COD), which might be included to have more values. This would also enable you to include the difference between fcc and bcc crystals. Unfortunately the database is written in python and therefore might be hard to include in a Matlab script, but I need to confess I did not try it. To make the program do everything for you, one might add a function to it, that calculates the position of the incoming and outgoing hole, but in my experiment I choose the most visible lines to calculate these, so the program would not be able to do that. One other thing annoyed me when using the program, sometimes you have traces of other elements in your data, which lead to lines that do not coincide

with the lattice constant, I would like the program to be able to throw these data out automatically, as I have to do it by hand at the moment. Also at the moment the program only include the data that is on the left side of the beam, when looking parallel to the sample wire. I figured this is the more precise data, but one could think about including the other data, too, if it looks good.

Marlon Nuske
December 2011