

Diffraction and imaging simulation Lab

Laboratory exercise 2

TFY4255 Materials Physics 2023

Lecturer : Antonius T. J. van Helvoort

Lab assistant : Syam Sadan

Email : syam.sadan@ntnu.no

Introduction

Fast Fourier transformations (FFTs) illustrate the basic ideas of reciprocal space and diffraction particularly well and are useful in image processing. In this lab, you will explore these phenomena. The main goals are that by using simple coding you visualize and better understand both the relation between real and reciprocal space, diffraction from different crystals and, theory presented in lecture & book. These are central parts of the course.

Unlike the other lab exercises, you can do this exercise in your own time, and the scheduled lab slot is as an hour where you can ask questions and discuss. Therefore, before the scheduled lab slot, we highly recommend that you try to perform the tasks by yourself or together with a lab partner. It is highly recommended to read the background literature [1].!

You need a basic understanding of python to execute the exercise. However, the focus should be on physical interpretations of the studied phenomena and the effects of different parameters. Therefore, you are provided with template code written in Python 3 via BB (python-file and a Jupyter notebook). These files and example images are also available via Jupyterhub [2], which requires no python installation on your own PC. You are free to choose the format you prefer, to write your own independent code, or to modify the provided code in any way you find convenient. If you write your own code or adapt the given code essentially, include the used code in an appendix in your report. The general report requirements are explained in a separate document. Be alert, what a cell does and what is plotted (ie. optimize, correct when required). You are also provided with some example images to work on. You are free to select images of your preference.

Diffraction from one dimensional crystals

Consider a one-dimensional (1D) crystal with atoms centered at $R_n = na$, , for atom $n \in N$, where N is the total number of atoms. The electron density of each atom can be modelled by a Gaussian with amplitude A_n and width σ_n . The total **electron density**, $\rho(x)$, is then given by

$$\rho(x) = A_n \sum_{n=1}^{n=N} e^{\left(-\frac{(x-R_n)^2}{2\sigma_n^2}\right)} \quad (1)$$

For kinematical diffraction, the scattering amplitude, $F(q)$, is given as the FT of the electron density of the crystal,

$$F(q) = FT(\rho(x)) \quad (2)$$

The diffraction pattern (or diffractogram) of the crystal shows the distribution of scattered intensity. The **intensity** is described by

$$I(q) \propto |F(q)|^2 \quad (3)$$

The intensity of the peaks in the diffraction pattern scales as $I(q) \sim N^2$, while their width is inversely proportional to the number of atoms ($\propto \frac{1}{N}$). For a perfect, endless crystal, the diffraction pattern consists of δ -functions located at the reciprocal lattice points.

If the crystal has a **monoatomic basis**, all atoms have the same atomic number, and the electron density, ρ , will have the same magnitude, A_n , and width, σ_n , for all atoms, i.e. for all n . A **diatomic basis** on the other hand, consists of two different atom types, which can be modelled for instance by letting the amplitudes, A_n , and/or widths, σ_n , of the Gaussians take one value for even n and another for odd n .

At finite temperatures, the atoms are moving about their lattice positions, which can affect the intensity distribution in the diffraction pattern. A simple way of including **thermal motion** in the model, is to add a random number $\alpha_n \in (-1,1)$ that is multiplied by a factor ε to each atom position, so that

$$R_n = na + \alpha_n \varepsilon \quad (4)$$

Statistically disordered (amorphous) materials with no long-range order, e.g. glasses, can be modelled by letting each atom position vary cumulatively with respect to the neighbouring atom positions, rather than with respect to an invisible ideal lattice. The atom positions in such a model are given by

$$R_n - R_{n-1} = a + \alpha_n \varepsilon_g \quad (5)$$

where ε_g determines the upper and lower limits of the positional fluctuations. Note the difference between the structures described by (4) and (5). The varying atom positions and electron densities are reflected in the observed intensities. You can alter structural parameters (eg. model, A_n , σ_n , α_n , N) and related them to calculated variations in $I(q)$.

Fast Fourier transformations and imaging

FFTs are useful for illustrating diffraction. The FFTs of images of objects are closely related to the diffraction patterns obtained from the same objects. Computing an FFT of an image is a mathematical operation where the image is expressed as weighted periodic functions with specific frequency and directions, instead of coordinates (or pixels) with intensity values. In analogy to microscopy, the FT works as a lens, and oppositely, a lens can be regarded as a Fourier transformer.

Furthermore, FFTs are useful for image processing. Image processing with FFTs involves computing and modifying this FFT of an image, before computing the inverse FFT (IFFT). This corresponds to mapping the image in the frequency domain (reciprocal space) and modifying this, before back-projecting the power spectrum into the spatial domain (real space).

Image filtering is image processing where certain aspects of the image are emphasised. Masking is the simplest case, in which image intensities are thresholded based on their value or by their coordinates. Placing a mask in the frequency domain of an image, can be compared to placing an aperture in the diffraction pattern of an object studied in a microscope. For example, in a transmission electron microscope, different aperture positions in reciprocal space can be used to obtain bright-field or dark-field images.

The information far out in reciprocal space corresponds to features in the image with small real-space periodicities, i.e. details in the input image (or object). An image can be noise-reduced by using a narrow mask, i.e. a low-pass filter, in reciprocal space. Noise-reduction competes with image blurring, and one should be careful not to introduce artefacts or misinterpret images when performing such image processing.

Lab tasks

Your task is, using code provided, to understand, demonstrate and visualize diffraction and image processing by using FFTs, for the cases presented below. Be creative and play with different structural parameters for given models, to gain a better understanding of the main concepts. Be alert what actual is computed and plotted.

1. Diffraction from one-dimensional crystals (Aim: link $\rho(x)$ and $I(q)$)

Illustrate and explain the differences between the following models and especially their corresponding simulated diffraction patterns; i) monoatomic basis, ii) diatomic basis, iii) disordered material (glass) and iv) material with thermal motion. You need to show both real and reciprocal, discuss the intensity distributions in reciprocal space and relate them back to charge densities in real space for structural parameters you investigated.

2. FFTs and imaging (Aim: link diffraction and imaging)

By using your own images and/or the provided example images, illustrate and explain

- (i) the appearance of the simulated diffraction patterns of differently shaped objects (e.g. triangles, squares, circles and periodic structures, own figures/photos),
- (ii) image filtering, noise-reduction and blurring by masking certain parts of frequency space and generating inverse fourier transform (IFT)

3. Diffraction from two-dimensional crystals

Illustrate and explain the differences between the simulated diffraction patterns of different models by extending the features/observations from 1D to 2D lattice.

Recommended literature:

[1] Aubert, E. & Lecomte, C. Illustrated Fourier transforms for crystallography. Journal of Applied Crystallography 40, 1153–1165 (2007).

URL <http://journals.iucr.org/j/issues/2007/06/00/kk5014/kk5014.pdf>.

[2] Jupyterhub (login required): <https://tinyurl.com/TFY4255-lab2-code>

or Download notebook or .py and data from <https://github.com/HelvoortTon/TFY4255-lab2>