Computing 2: Computational Physics: Final project report

Zofia Wilk

University of Leeds

March 2024

1 Plots

Firstly, the read_diffraction_data function extracted the metadata, intensity, $sin(\phi)$ and $sin(\theta)$ values from the file. All the features - "Noisy Data", "Variable Wavelength", "As projected", "Variable Metadata" and "Aperture shape selected at random" were included in the project. The data was then plotted as shown in Figure 1 below:

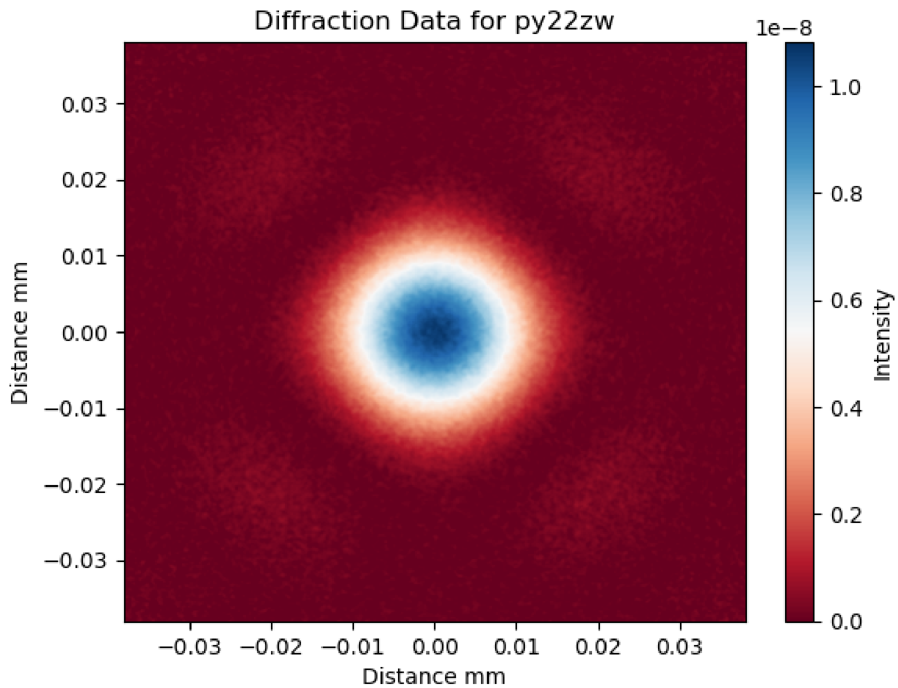

Figure 1: Plot of diffraction data extracted from the file.

The intensity slice was then taken through $\sin(\phi) = 0$. As the aperture was randomly selected, fits for square, rectangle, diamond, and circle were performed using the **curve_fit** function. The plot of this fit can be seen in Figure 2. To determine the shape of the aperture from this fitting, the value of chi-squared for all shapes was calculated. The shape with the lowest chi-squared was a diamond, so the diagonal slice was chosen as a cross-section for the second plot. The plot for a slice through the diagonal can be seen in Figure 3. The best-fit shape was found to be square, confirming the initial results that the shape of the aperture is a diamond. The best-fit shapes are marked with an asterisk $*$ in the legend of both graphs.

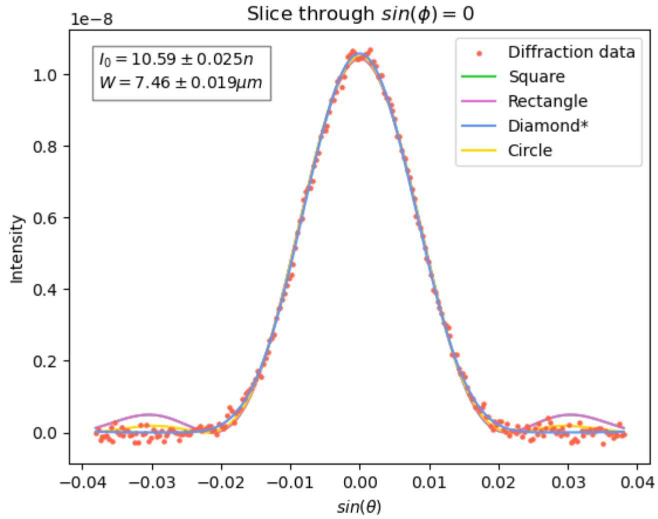

Figure 2: Plot of a slice of data through $sin(\phi) = 0$. The best-fit aperture is marked with $*$.

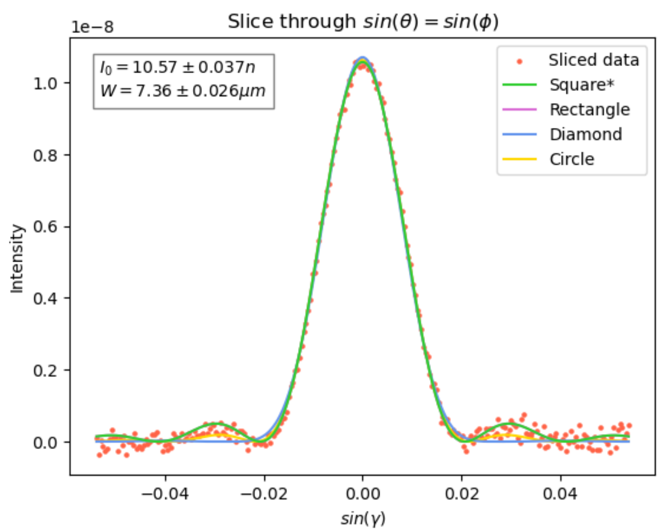

Figure 3: Plot of a diagonal slice of data (through $sin(\theta) = sin(\phi)$). The best-fit aperture for this cross-section is marked with $*$.

2 Numerical data

The aperture shape was determined based on the lowest chi-squared value among all the possible fittings for the first slice and either dimension difference if the shape was 'square' or 'rectangle', or chi-squared for 'circle' and 'square' from the second slice if the shape was 'circle' or 'diamond'. The code was tested 7 times for different practice data and accurately returned the correct shape each time. The dimensions and intensity per area, along with their corresponding errors, were calculated from the *pcov* and *perr* values outputted from the curve_fit using **calculate_anwers** function for the determined aperture shape. The final results from the fitting for the final shape can be seen in Table 1.

Results	
Quantity	Value
Aperture shape	Diamond
Dimension 1	$7.46 \pm 0.019\mu$m
Dimension 2	$7.36 \pm 0.026\mu$m
Intensity $I_0/area$	192.86 ± 0.97 Wm^{-2}

Table 1: Table showing the results of the analysis for the project.

3 Flow Diagram

The code starts with a **proceess_diffraction_data** function that extracts the necessary information from the given file in the **filename**. It returns the dictionary of metadata, intensity, sin_phi, and sin_theta. These variables are then used in **plot_diffraction_data** function, and the output is as shown in Figure 1.

Moving to tasks 3 and 4, separate slicing is performed, one through $sin(\phi) = 0$, and the second through $sin(\theta) = 0$ or $sin(\theta) = sin(\phi)$ depending on the shape found after the first slice. As the aperture shape is selected at random, the fitting functions for all possible shapes are defined as **intensity_square**, **intensity_rectangle**, **intensity_diamond** and **intensity_circle**. The **curve_fit** is then used to extract optimal parameters from the fitting functions for all the shapes for the $sin(\phi) = 0$ slice, called popt_1_shape, pcov_1_shape on the diagram for simplicity. All possible fits are then plotted and are shown in Figure 2. If the shape is random, to determine the actual shape, the **chi_squared** function is first defined and called for fittings of all shapes. The lowest value among the chi-squares is then found, and the corresponding shape is assigned. On the other hand, if the shape of the aperture is known, the code automatically extracts the value from the metadata. Following that, if the shape found from $sin(\phi) = 0$ slice is a 'rectangle' or a 'square', the next slice of data is chosen to be through $sin(\theta) = 0$, and fittings for 'rectangle' or 'square' are performed to find the second dimension of the aperture. If the dimensions differ by more than 1 micron, then the shape is a 'rectangle'; if not, the shape is a 'square'. However, if the shape is either 'diamond' or 'square', the diagonal slice (through $sin(\theta) = sin(\phi)$ or $sin(\sqrt{\phi^2 + \theta^2})$) is taken. The **curve_fit** values are then calculated for all shapes and plotted as shown in Figure 2. If the shape is a diamond, the diagonally sliced data should fit 'square' more, and if not, it should stay the same and fit 'circle'. To check that, chi-square values are calculated for the square and circle, and the final shape is chosen based on which had the smaller chi-squared between those two.

Lastly, the final results for the project are calculated using **calculate_answers** function which utilizes fitted parameters for the final shape to return **dim_1, dim_1_err, dim_2, dim_2_err, I0_per_A, I0_per_A_err, I0_per_A_err**. Inside this function, the **calculate_I0_per_A_err** function was defined and called to calculate the uncertainty of the Intensity per Area. Additionally, the code checks if the dimensions are within the allowed range of 1 to 40 microns. If they are not, a ValueError is raised to inform of the incorrect results.

The returned values from **calculate_answers** are then used to create a dictionary inside the **ProcessData** function. This returns the final results for the project. The Flow Diagram of the whole process can be seen in Figure 4 below. For simplicity **popt_1_square, popt_1_rectangle, popt_1_diamond** and **popt_1_cirlce** are marked as **popt_1_shape**, same for **popt_2_shape**.

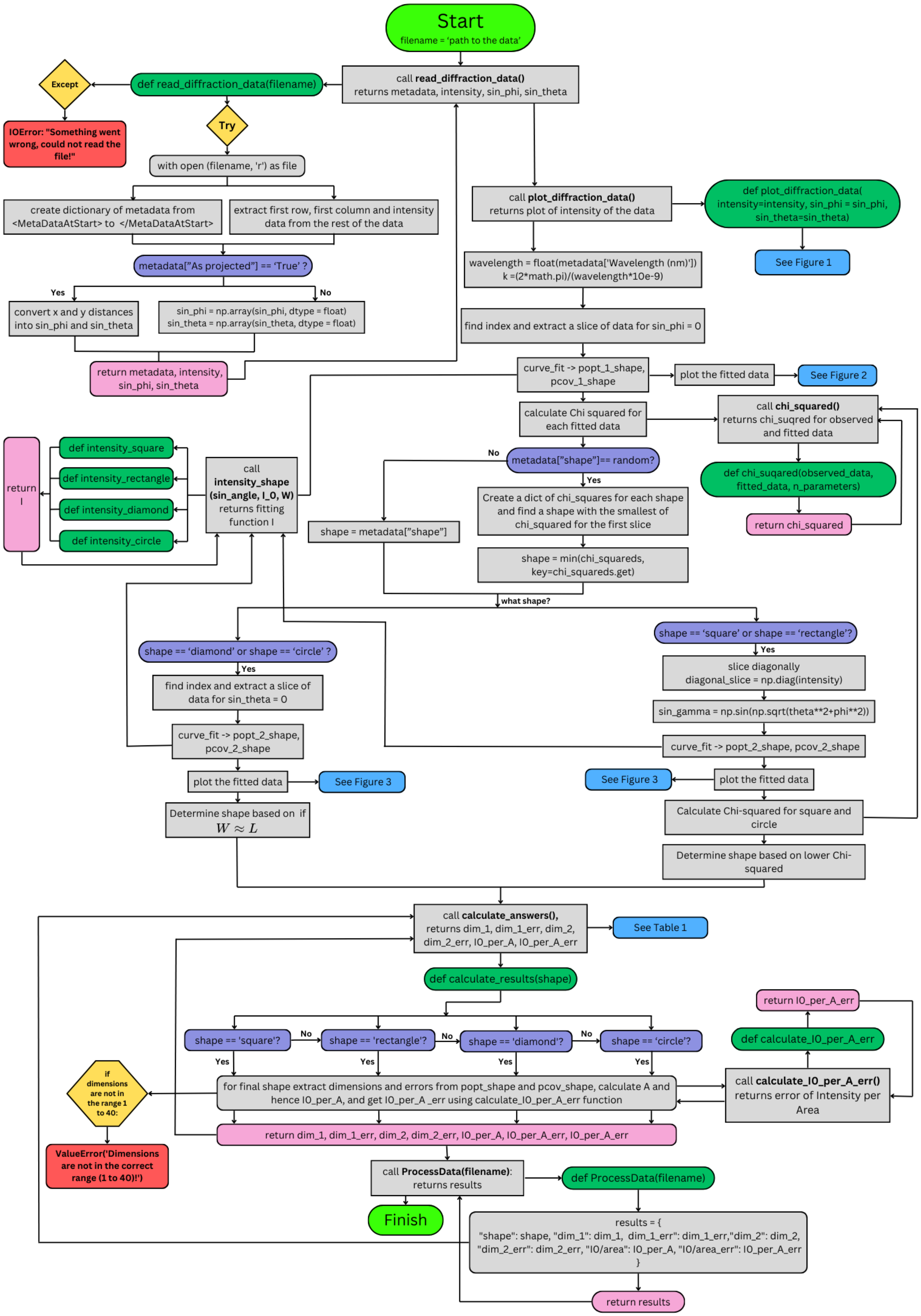

Figure 4: The Flow Diagram for the project.