

# Developing a GUI for thin film characterization

Alexandre Gusmão<sup>1,a</sup>

<sup>1</sup>Faculdade de Ciências da Universidade de Lisboa, Lisboa, Portugal

Project Supervisors: R. Nunes, T. Sousa, P. Teubig

October 17, 2023

**Abstract.** An article describing the methodology, workflow, creation and methods of a Graphic User Interface (GUI) that facilitates the characterization of thin films, using the Alpha Energy Loss method. This GUI was created in Python, using the Tkinter library. The calculations are based on the procedures used by the NUC-RIA team to obtain the thickness of a film, and are shown in this article. Simple algorithms were developed to detect the peaks of the energy loss data and will likewise be addressed.

**KEYWORDS:** Thin Films, Characterization, AEL, Python, Tkinter

## 1 Introduction

The one thing all fundamental nuclear reaction studies and production of radioactive beams have in common is that they require a target. These thin films are made from the required material with their thickness range from a few nanometers to micrometers depending on the technical requirements of the experiment.

### 1.1 Characterization of Thin Films

A very important aspect of the production of the targets is that their thickness need to be properly assessed. There are a few procedure available for the estimation of the thickness for films. One is the Rutherford Back-Scattering (RBS) method, another, and the focus of this work, is the Alpha Energy Loss (AEL) method [1]. With AEL, the thickness of a thin film is estimated by calculating the kinetic energy of the particles with a thin film between the detector and the radiation source, and comparing it with the kinetic energy of the decay chain reaction in a vacuum. The alpha particles will loose kinetic energy as they pass through the film, resulting in measurable energy loss. This relation can be approximated by the following equation, known as the stopping power:

$$\Delta x \approx \int_{E_i}^{E_f} - \left( \frac{dE}{dx} \right)^{-1} dx \quad (1)$$

Different stopping power values for various materials over an energy range from 1 keV to 1 GeV of alpha particles are available in the NIST ASTAR database [2]. Since the values in use are discrete, an approximate equation 1 as a sum with an integration step is applied, as following:

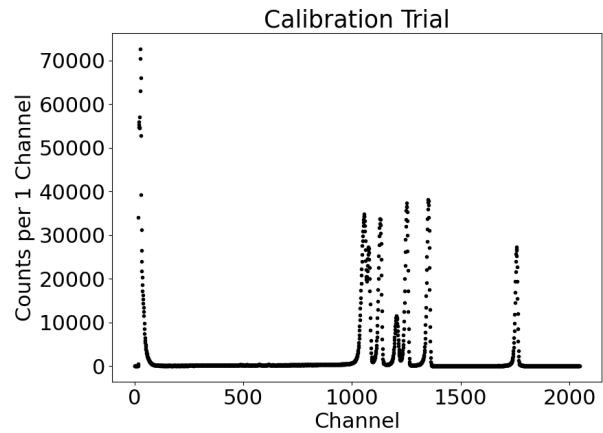
$$\Delta x \approx \sum_{E_k=E_i}^{E_f} \frac{\delta E}{S(E_k)} \quad (2)$$

Where  $\delta E = E_{k+1} - E_k$ .

This last approximation allows the calculation of the thickness of a given film using the AEL method, which is a non destructive, inexpensive method of thickness estimation. As such, this is the method chosen to calculate

the approximated thickness of the film, in the background of the GUI, as equation 2 is very easily implemented in code. A full description of the analysis performed by the NUC-RIA group follows:

1. Plot the experimental data for the calibration run (the trial with no thin film) and find the peaks in the graph, that correspond to the energy of the alpha particles in the decay chain.



**Figure 1.** Plot representing the decay reaction of Uranium-232, in a calibration run.

2. Relate the energy values of the element's decay chain with the channels of the peaks, in a linear regression. The result will be used to calibrate the runs with a thin film.
3. Repeat the step for the experimental data of a run with a thin film. Search for the peaks, and calibrate them with the linear regression.
4. Establish a range that results from the energy values of the calibrated channels to the element's decay chain, for each peak. The results will yield a range, that contain the starting and finishing values of the sum in equation 2.

<sup>a</sup>e-mail: alex\_vng@hotmail.com

- Repeat the process for every peak the user deems fit. Average all the values obtained from the previous step. The final result is the approximation to the thickness of the film.

## 1.2 Motivation for a Graphic User Interface

While the necessary calculations to derive an approximate thickness are straightforward, the overall process is strenuous and time-consuming. In particular, obtaining the correct channels with the highest counts and the corresponding Stopping Power range, prove to be the most downbeat steps. Therefore, a GUI that characterizes thin films by the AEL method must perform these tasks swiftly and optimally, as well as keep most of the remainder calculations in background. In this fashion, user errors during analysis are minimized, while the user needs to retain a certain degree of control over the parameters. Therefore, the program should possess:

- An internal database of the stopping powers for various materials, as well as the values of the radiation sources in use.
- Algorithms that in an automatic or semi-automatically fashion detect the correct channels for the calculations.
- Algorithms that perform linear regressions, as well as equation 2 for different units.

Since an automatic analysis can be prone to various errors (i.e. wrong identification of peaks), a balance between user input and the background calculations must be achieved. Besides the selection of the databases and the management of the original data files, the user should have power over which values to consider for a linear regression, as well as which of these are used to calibrate a trial with a thin film.

## 2 ARC-TF

Keeping all these factors in consideration, a program, ARC-TF or, Alpha energy loss and Rutherford back-scattering for Characterization of Thin Films, was designed in Python 3.9, with the non-native libraries of Pillow and Matplotlib and the native libraries of Tkinter, Webbrowser, Math, OS, Shutil and Ctypes. It also includes the energy values of two radiation sources the group uses, Uranium-232 and Radium-226, as well as the stopping power values, ranging from 4 MeV to 8 MeV, of the materials Aluminum, Gold, Lead, PMMA, and Tin, as a native database.

### 2.1 Algorithms

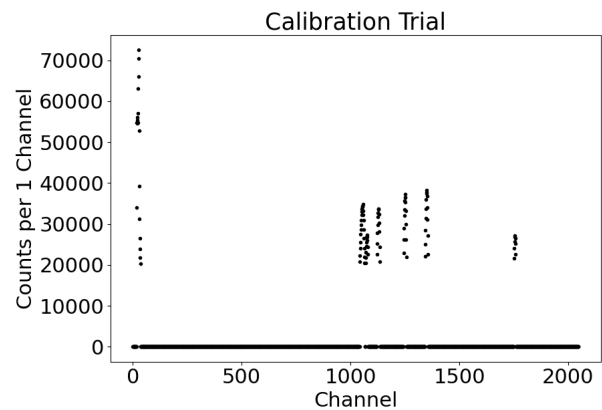
ARC-TF contains 4 major algorithms with three different functions:

- Two different algorithms that search for the channels with most counts in a spectrum. These are the Threshold Input Algorithm and the Manual Selection Algorithm.
- An algorithm that performs a linear regression and calculates the uncertainty for the intersect and the slope.
- An algorithm that takes in all the information from the previous algorithms, and calculates the approximate thickness of the film, using equation 2.

These algorithms will be explained in depth in the following sections.

#### 2.1.1 Threshold Input Algorithm

This algorithm detects the various local channels with the highest counts. After uploading the experimental results from the detector, the user can input a threshold value. With this value, the program will only read the experimental results above this value, and ignore all the remainder. An example of how the algorithm treats the data follows:



**Figure 2.** Plot demonstrating how the Threshold Input Algorithm views the uploaded data.

In Figure 2, after the separation of the counts, the program will then detect the maximum of each curve. However this causes two problems: Firstly, the algorithm will treat the low energy noise as a peak and consider certain channels as values of interest. Secondly, some of these curves will contain a few points that might fall below the threshold, probably due to systematic errors. These problems become especially clear when compared with Figure 1.

The solution implemented for both problems is the same: Two variables that mitigate the effects. The first will only take into account values beyond a certain channel. This will solve the low energy noise points that would be detected. The variable is configured within the code with the value of  $Channel = 100$ . As for the other variable, if the algorithm registers two points whose channels differ only by a value of a certain amount,  $Channel_{k+1} - Channel_k < 35$ , it will only keep the one with most counts. Both of these values can be altered in the Settings Menu of the program, and their function is to keep the algorithm more exact, in order to minimize the intervention of the user.

After the peaks are detected, the program stores the Counts and the Channels of the points in a temporary file.

### 2.1.2 Manual Selection Algorithm

This algorithm, like the one before it, detects the various points. However, unlike the Threshold Input, there is no input value from the user and it does not necessarily detect local maximums. Instead, the user can simply right click with his mouse on the graphic, to detect the nearest points. This means that the user has freedom to select the maximum values, but also other points. The algorithm is simple, only using a least square difference to check which points to save in a temporary file:

$$Point = \min \left( \sqrt{(y_k - M_y)^2 + (x_k - M_x)^2} \right) \quad (3)$$

Where  $y_k$  and  $x_k$  are the experimental values, corresponding to the counts and channel from  $k = 1$  up to  $k = n$  respectively.  $M_y$  and  $M_x$  are the counts and channels the user clicked on. The algorithm then configures any point to a temporary file, treating it as a local maximum. It is a semi-automatic algorithm that can be used in junction with the Threshold Input Algorithm interchangeably. The exactness of the values depends only on where the user clicks.

### 2.1.3 The Linear Regression Algorithm

This algorithm performs a simple linear regression. It is included in the article simply to showcase the nature of the linear regression method, and how the uncertainty of the parameters are calculated. It starts by sorting the results of the channels detected by the previous two algorithms and the values of decay energy of the radiation source. This does not create any error for the linear regression since the values have an ascending order. In fact, since the user can acquire the channels of the spectrum in any order, it is an imperative step to perform. After this step, the equations for the slope ( $m$ ), and intersect ( $b$ ), follow:

$$m = \frac{\sum_{k=0}^n (x_k - \bar{x})(y_k - \bar{y})}{\sum_{k=0}^n (x_k - \bar{x})^2} \quad (4)$$

$$b = \bar{y} - m\bar{x} \quad (5)$$

Where  $y_k$  are the values of the decay energy of the alpha particles source and  $x_k$  are the channels detected. Because there is no method in the program that calculates the uncertainty of the channels from the experimental results, the uncertainty of the slope and intersect are acquired from the following equations:

$$\sigma_m^2 = \frac{n\sigma^2}{n \sum_{k=0}^n x_k^2 - \left( \sum_{k=0}^n x_k \right)^2} \quad (6)$$

$$\sigma_b^2 = \frac{\sigma^2 \left( \sum_{k=0}^n x_k \right)^2}{n \sum_{k=0}^n x_k^2 - \left( \sum_{k=0}^n x_k \right)^2} \quad (7)$$

Where  $\sigma$  comes from the following equation:

$$\sigma^2 = \frac{\sum_{k=0}^n (y_k - mx_k - b)}{n - 2} \quad (8)$$

The procedure for calculating the uncertainty, until a method to obtain the uncertainty of the experimental data itself is implemented, remains in a generalized form. This has the consequence of a larger uncertainty, but suffices for preliminary results, and an initial version of ARC-TF.

### 2.1.4 The Thickness Approximation Algorithm

The final calculation receives data from the previous algorithms and condenses it for equation 2.

The algorithm needs to apply the calibration, using the linear regression results, to the channels of the spectrum with a thin film. Due to the nature of the experimental setup of the Alpha Energy Loss, it is necessary to change the vacuum chamber. This might uncalibrate the setup. For precaution, new calibration runs were performed. Should the linear regression deviate significantly from the original calibration, it indicates that the experimental data might not be exact. However, if the deviation is neglectable, the user has the option to select both, (or more), linear regressions from the different trials. For the algorithm, this is possible for any number of calibration trials the user wishes to select. The algorithm averages all the linear regression values, and applies the final results to the channels detected by the first two algorithms on the spectrum of the thin film.

The next step consists on finding the stopping power range, that will be used in equation 2. For this, the program checks the decay energies used in the first calibration trial the user has selected for the linear regressions. It will only consider the energy values of this calibration trial. These values become  $E_f$  of equation 2, while the channels, after the linear regression is applied to them, become  $E_k$  with  $k = 1$ . With this range, equation 2 is computed, and with the internal database of a few materials, an approximate thickness is calculated. This is done for every channel corresponding to a peak. After this step, an average of the thickness is obtained, while its uncertainty is computed using the following equation:

$$\delta x = \frac{\sigma}{\sqrt{n - 1}} \quad (9)$$

Where  $n$  corresponds to the number of thicknesses calculated and  $\sigma$  to the standard deviation of the samples.

The results are then presented in the selected units.

## 3 Workflow

The program has three types of tabs: two analysis tabs, Calibration Tab and the Material Tab and the Final Results Tab. The analysis tabs, process the data and calculate linear regressions and the thicknesses of thin films. The user can create and delete these two types of tabs freely, unlike the Final Results Tab, which is special, as it can not be deleted and exhibits all the results calculated by the analysis tabs. An image detailing the schematic of the workflow is presented.

## ARC-TF Workflow Schematic

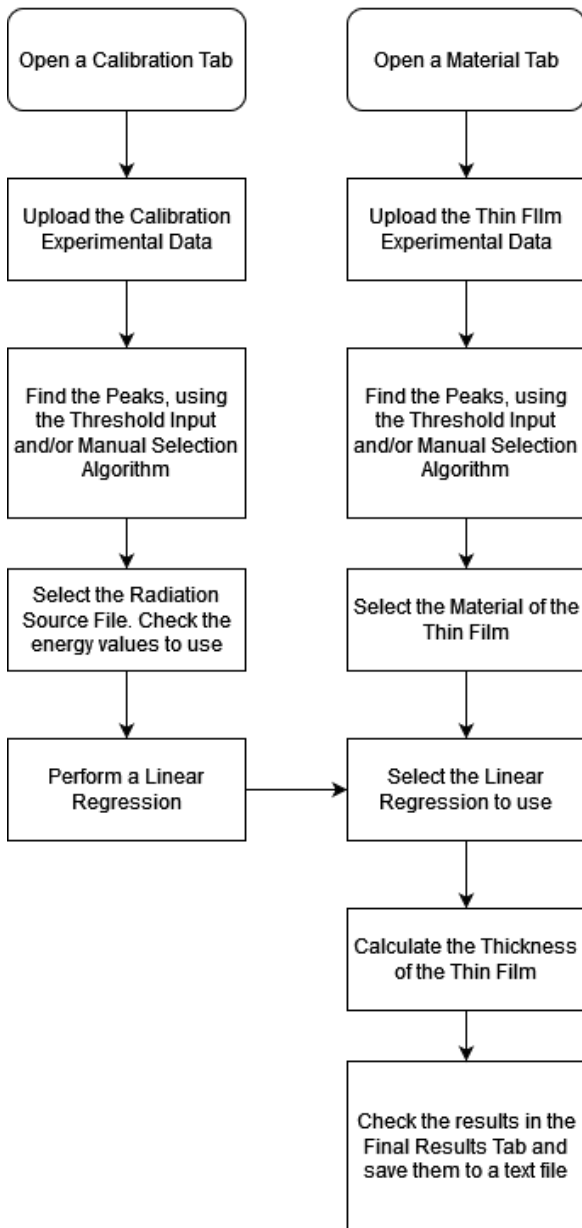


Figure 3. Depiction the workflow of ARC-TF.

### 3.1 Calibration Tab

The program automatically starts with one calibration tab open. However, the user has the option to open more calibration tabs. Here, the user can click the File button on the menu bar, and click the first option to upload the experimental data that is saved in the computer. ARC-TF will automatically plot this data and show it embedded in the GUI. From here, the user can detect the peaks, selecting the analysis algorithms first, up to a maximum of 15 values. The corresponding channels and counts will appear on screen at the top right corner. The user can reset all these values, or uncheck a selection of them. If

unchecked, the user needs to clear these values. Following this analysis, the user can select the radiation source. Once again, the user has the possibility to check/uncheck the energy values that the interface will use for the both the linear regression, and the final energy step from equation 2. The program will only perform the linear regression if

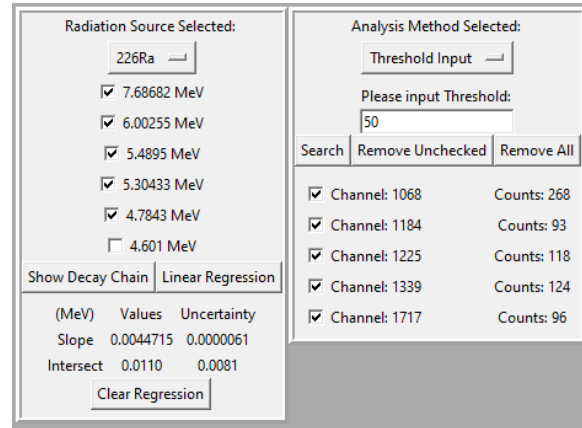


Figure 4. Results of the Calibration Tab, after selecting the values of energy to use, and analyzing the experimental data.

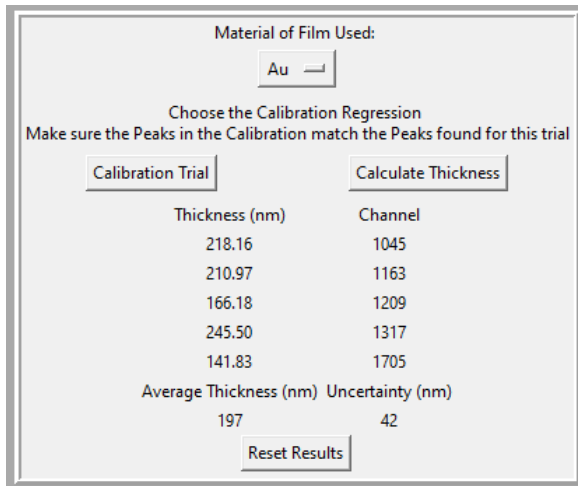
the number of the checked energy values match the number of peaks detected and checked. After these values are confirmed, the user can perform a linear regression. The results will appear in the calibration trial, and in the final results tab.

### 3.2 Material Tab

The user can open a material tab at any step of the analysis process. The first few steps, as seen in figure 3, are equal to the calibration tab. The user uploads the data, detects the peaks using the analysis algorithms. Here, the user also needs to make an appropriate selection of the channels intended for use. It is important that the user selects the peaks such that they match the ones selected in the calibration trial. Following this step, the user can then select the material of the thin film. Afterwards, all that is left is to choose one, or more, calibration trials. The program will only use the energy from the radiation sources of the first calibration trial. This is why the choice of peaks is of great importance. Finally, the user can calculate the thickness by clicking the button (see figure 5). The results will appear below the button, and in the final results tab.

### 3.3 Other Features

Besides the various tools explained previously, ARC-TF also has a few other features that facilitate the analysis. The program has the option to delete the various instances of data, so the user can restart the analysis from a specific point. Likewise, the user can clear the entire data from a tab. The GUI has a settings menu, where the user can select the units to use. These vary for the units of the linear regression, which can be in keV or MeV, and for the



**Figure 5.** Results of the Material type, after selecting the Linear Regression, material and analyzing the data.

units of thickness, which can be in: nm,  $\mu\text{m}$ ,  $\mu\text{g cm}^{-2}$  or  $10^{15}$  atoms  $\text{cm}^{-2}$ . These options can be applied to the entire program, or to a specific tab. This makes analysis for various thin films of different orders of magnitude possible. Furthermore, the user can create and delete tabs and add or remove material or radiation source files from the database. For the calibration trial, there is also the option for viewing an image detailing the the decay reaction of the selected radiation source. More images can be added or removed by the user. Besides these features, there is also an About button, which will direct the user to the GitHub repository of the GUI.

### 3.4 Limitations

ARC-TF is intended to be a simple and effective analysis tool, but naturally, a few limitations were considered. As of the publication of this article ARC-TF works only for Windows.

The program only reads MCA domain files when the user uploads the file that has the experimental data. Furthermore, the program expects that the experimental values start only at line 13. In case of the files of the database, it will only read TXT domain files. In case of the materials database, the file must have the relative atomic mass in the first line, followed by "|" and then, the density plus another "|". For the remaining lines, the TXT file must be organized with the energy value followed by "|", the corresponding stopping power, and another "|". In case of the radiation sources, the values need to be written in descending order of the values in MeV.

The Graphic User Interface was designed to work for any given range of energies. However, the built-in database has the stopping powers corresponding to the energy range of 4 MeV to 8 MeV. This interval was chosen due to the energy of the radiation sources the NUC-RIA

utilizes for their experiments. The incrementation of the range corresponds to 1 keV, as this is the lowest incrementation available in the ASTAR database from NIST. The

program includes the stopping powers for the materials of Aluminium, Gold, Lead, PMMA and Tin.

For the radiation sources of alpha particles, ARC-TF has the the values of Radium-226 and Uranium-232 in MeV. In both the material and radiation sources, it is possible to add new files externally to ARC-TF's file folder, or internally.

## 4 Conclusions and the future of ARC-TF

A mixture of physics knowledge and programming research, ARC-TF is the product that attempts to simplify the analysis of thin films characterized by the Alpha Energy Loss method. The product aims to be simple, fast, complete and structured, and as of the latest push in the repository, is fully functional to analyze thin films. This was a result of a summer internship at LIP, where I had first contact with the Python programming language and basic information on the production of thin films as well as nuclear physics. As for the future of ARC-TF, the first major step would be to make compatible versions for MAC and Linux. In case of additional features, although the Rutherford back-scattering is included in the name, there is not, as of yet, a way to analyze thin films with this method in the program. Besides this implementation, other features like AlfaMC, a simulation tool for predicting the Alpha Energy Loss spectrums, by Peralta, L., Louro, A., [3], and machine learning algorithms can and should be implemented to make this a powerful tool for analysis of thin films in various laboratories. ARC-TF is available here.

## Acknowledgements

Thanks to the NUC-RIA group for welcoming me, for the various opportunities I was presented and for the pleasant and learning environment. I would also like to extend my thanks to my supervisors, and to Ricardo Pires who accompanied me for the entire journey.

## References

- [1] P.S. Morral, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **613**, 423 (2009)
- [2] NIST, *Stopping Power for Helium Ions*, <https://physics.nist.gov/PhysRefData/Star/Text/ASTAR.html>
- [3] L. Peralta, A. Louro, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **737**, 163 (2014)

