Introduction

This helpbook is an additional guide for the “nk-simulation” software. This software was developed for calculating nk-data and the layer thickness of thin films on monolayer substrates. The determination is based on parameter optimization of the dielectric function of the thin films. This is achieved through numerical approximation of experimentally determined Reflection/Transmission (RT) measurements and/or spectral ellipsometry (SE) measurements. Users can choose which measurement method to consider for numerical fitting. The dielectric function can be selected from various models for transparent (Cauchy, Sellmeier) and absorbing (Tauc-Lorentz, Drude) materials. Additionally, a Bruggeman roughness layer can be considered to model surface roughness.Installation

Download & Installation

To run the software, a Python environment must be available. The following Python packages must be installed beforehand:

- scipy
- numpy
- pyqt5
- pyqtgraph

To download the software you can clone the repository in Git BASH or simply download and unzip the whole folder on GitHub.

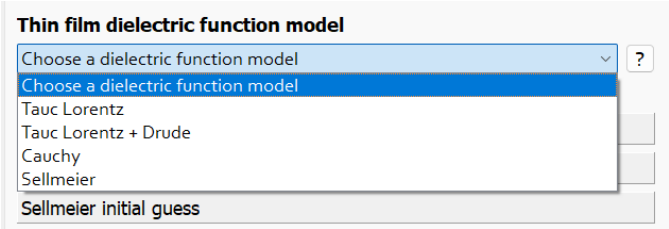
Getting started

To start the program, run the "GUI.py" script from the main folder. If the necessary Python packages have been installed successfully, the “MainWindow” and a “Console Output” should open. The software offers three main functionalities depending on which checkboxes are activated:

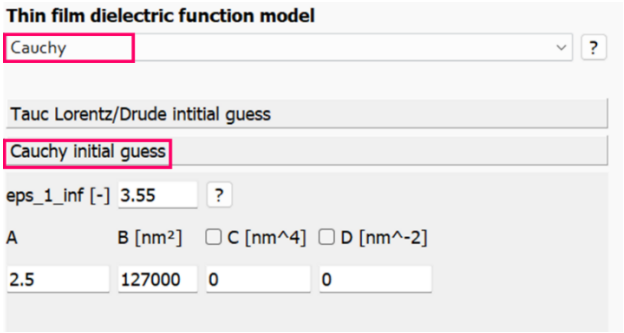
- No checkbox activated: Calculation of nk-data using the given dielectric model/parameters in “Model settings”
- Calculating RT/SE: In addition to nk-data, RT/SE spectra are calculated using the model parameters in "Model Settings" and the specifications of the RT/SE measurement in "RT settings" or "SE settings".
- Fitting RT/SE: Performs fitting to uploaded RT/SE data. The nk-data is then calculated from the optimized parameters.

Model Settings

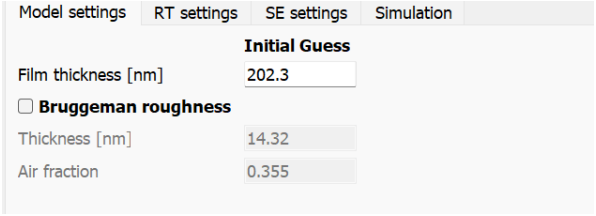
In the Model settings, the dielectric function used as the basis for calculating nk-data is specified. Users can choose from the available models.



Additionally, initial values for the chosen model are set. Only the model selected from the list is considered, regardless of the input initial values.



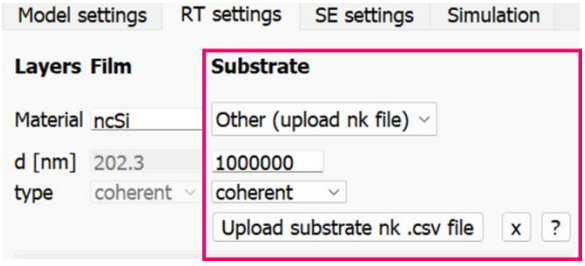
A Bruggeman roughness layer can also be included to describe surface roughness.



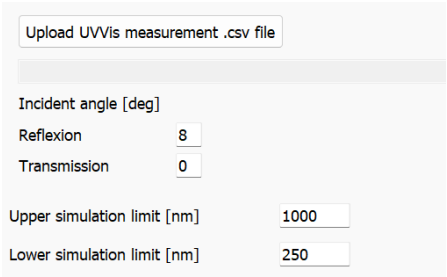
UVVis Settings

In the RT settings, the parameters for the R/T calculations are configured. This window is crucial when the 'Fitting RT' or 'Calculating RT' checkboxes are activated:

- **For Fitting RT and Calculating RT:** The nk-data of the substrate must be specified either by selecting from the available substrates or by uploading custom substrate nk-data. The thickness and coherence properties of the substrate are also chosen. These settings are mandatory for fitting/calculating the RT spectra.



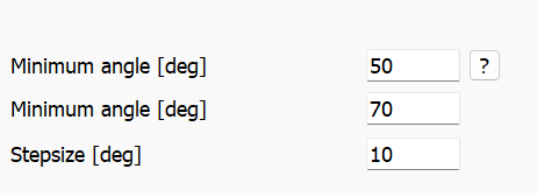
- **Only for Fitting RT:** Experimentally determined RT spectra must be uploaded by clicking the "UVVis measurement .csv file button." The incidence angle and the wavelength range for the simulation are also specified.



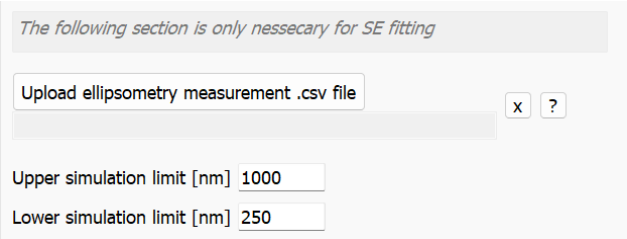
SE settings

In the SE settings, the parameters for calculating SE spectra are configured. This window is crucial when the 'Fitting SE' or 'Calculating SE' checkboxes are activated:

- **For Fitting SE and Calculating SE:** Similar to the RT Settings, a substrate must be specified by selecting from the available substrates or by uploading nk-data.
- **For Calculating SE:** While calculating SE the incident angles should be specified



- **Only for Fitting SE:** For fitting SE, a corresponding SE measurement series must be uploaded, and the wavelength range for the simulation must be specified.



Simulation Settings

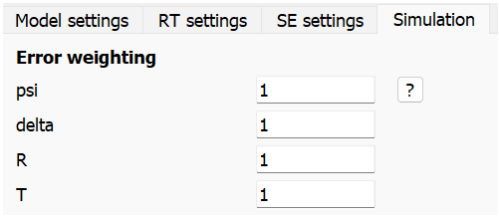
The simulation is performed by minimizing the weighted sum of squared errors using a Levenberg-Marquardt algorithm. The cost functions to be minimized are:

- **Fitting RT:** $F_{RT}(x) = \frac{1}{2N_\lambda}\sum_{t=1}^{N_\lambda}\left[\alpha_R \cdot \left(\frac{R_t^{fit}-R_t^e}{\max(R^e)}\right)^2 + \alpha_T \cdot \left(\frac{T_t^{fit}-T_t^e}{\max(T^e)}\right)^2\right]$

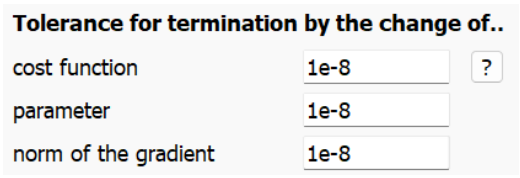
- **Fitting SE:** $F_{SE}(x) = \frac{1}{2N_\lambda}\sum_{t=1}^{N_\lambda}\sum_{j=1}^{N_\theta}\left[\alpha_\psi \cdot \left(\frac{\psi_t^{fit}-\psi_t^e}{\max(\psi^e)}\right)^2 + \alpha_\Delta \cdot \left(\frac{\Delta_t^{fit}-\Delta_t^e}{\max(\Delta^e)}\right)^2\right]$

- **Fitting RT and SE:** $F_{SE+RT}(x) = F_{RT} + F_{SE}$

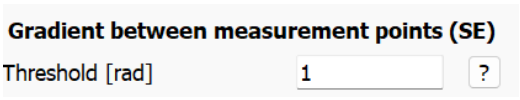
The error weighting factors α_i can be set in de Simulation settings. Especially when fitting RT and SE simultaneously you should consider adapting $\alpha_{R/T}$ when multiple SE data sets with different incident angles are fitted at once. This should help to consider the RT error with a similar weight even though there is much more SE data influencing the total cost function. Since the datapoints are interpolated over the given wavelength range it is not necessary to take into account differences between the spectre length of th RT and SE measurement.



Additionally, the termination criteria for the optimization can be defined based on the cost function (python argument “ftol”), the optimized parameters (“xtol”) and the norm of the gradient (“gtol”). The tolerance criteria refer to the change in the respective quantity between two iteration steps.



For numerical stability during SE fitting, a criterion for the slope between two adjacent data points is provided. This ensures that the nearly discontinuous behaviour of the ellipsometry angles, which can occur particularly at high incidence angles, does not disrupt the numerical approximation. The gradient between two adjacent data points is checked and excluded from error minimization if it exceeds the specified threshold.

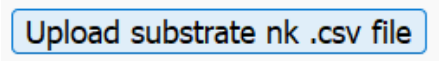


It is proposed to not change the default settings in simulation unless you are sure what you are doing.

Data upload

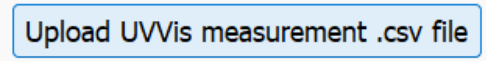
For uploading substrate nk-files and RT/SE data files there need to be a specified formate. Generally:

The uploaded files need to be .csv files semicolon-separated (;):



Requires a .csv file with 3 columns.
Example file: */CSV for python/Substrate/Corning Glass*

[Wavelength in nm] ; [n] ; [k]



Requires a .csv file with 3 columns.
Example file: */CSV for python/RT_example_ncSiOx*

[Wavelength in nm] ; [R] ; [T]

The first row contains the initial angles of the measurement. From the 2nd row onwards the file contains the actual measurement.

Example file: */CSV for python /CSE_example_ncSiOX_50_60_70*

[Angle 1 in degree] ; [Angle 2 in degree] ; [Angle 2 in degree] ;...

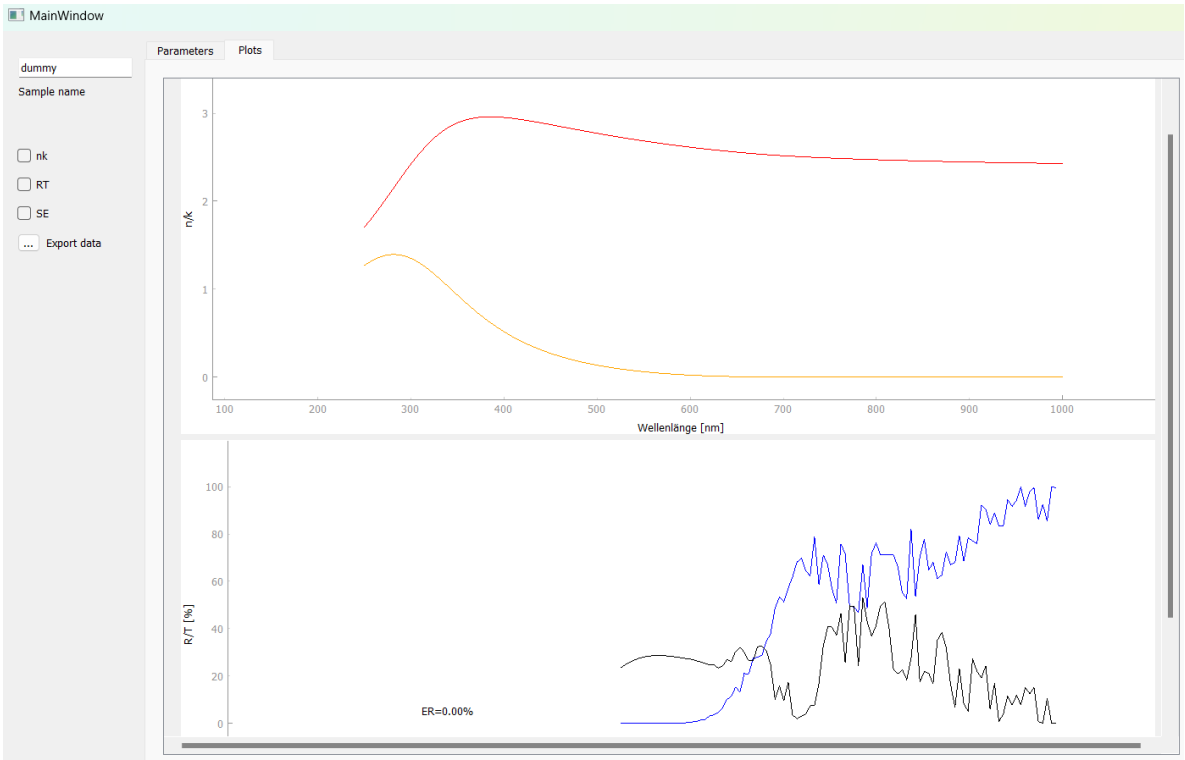
[Wavelength in nm] ; [psi angle 1 in rad] ; [delta angle 1 in rad] ; [psi angle 2 in rad] ; ...

Result window

The Result window contains two tabs:

The first tab shows the parameter values. If a simulation was performed, these are the results of the parameter optimization. If only a calculation without optimization was done, the user's inputs in the Model Settings are displayed.

The next tab shows the nk-data, R/T-data, and psi/Delta-data plotted over the wavelength, depending on the selection. The visible plots can be exported as .csv files. To do this, select the desired data and choose a save path by clicking the 'export data' button.



Console Window

The Console Output window is used to monitor the simulation progress. Every 10th iteration step displays the current parameters. If the simulation needs to be stopped early, the "Stop Calculation" button can be pressed. The console window, like this handbook, can always be accessed from the main window.