A Python-Based Measurement Model Toolbox for Impedance Spectroscopy

William Watson and Mark Orazem
University of Florida

August 4, 2020

Dedicated to our families.

Contents

List of Figures

List of Tables

Preface

The measurement model described in this guide, and this guide itself, are based on a program and guide developed in our group in the 1990s. Work on the original measurement model began in 1989 and culminated in a graphical user interface written primarily in Matlab® that interfaced with Fortran executables. Our original measurement model program was not made generally available because each new version of Matlab® broke a part of the program, and a part of the FORTRAN code was not ours to share. The new code was written from the ground up in Python, although some portions of the back-end libraries, such as `numpy`, use C.

The measurement model was developed in our group the early 1990s as a means to quantify the error structure of electrochemical impedance spectroscopy measurements. The work represented an extension of an approach developed by García-Rubio for optical characterizations[7–9] to spectra associated with electrochemi8cal impedance measurements. In collaboration with García-Rubio, our group demonstrated that a Voigt series can provide an adequate fit to impedance spectra and could therefore serve as a generalized measurement model.[1] By filtering lack of replication of similar impedance spectra, the measurement model was used to identify the standard deviation representing the stochastic error of impedance measurements.[2] The measurement model was also used to identify the frequency range of impedance measurements that was consistent with the Kramers–Kronig relations.[10,3]

Significant work was expended in the mid 1990s to the mid 2000s to verify the stochastic error structure identified by the measurement model. Durbha et al.[11] showed that the statistical properties of the error structure identified by the measurement model were consistent with the propagation of errors through the Kramers–Kronig relations. Carson et al.[12,13] showed, by numerical emulation of impedance instruments, that time-domain noise propagated into the frequency domain had properties similar to those obtained by the measurement model analysis. Shukla et al.[14] used both rational[15,16] and Voigt measurement models to show that the stochastic error structure identified was independent of the form of the measurement model used.

While instrument vendors often provide a means of assessing consistency of impedance data to the Kramers–Kronig relations, these approaches are generally based on the linear regression approach pioneered by Boukamp.[17] You et al.[18] showed that

the Voigt measurement model presented in this book was more sensitive to failures of causality associated with nonstationary behavior than was the linear regression approach.

The measurement model also has application for preliminary characterization of impedance data. Orazem et al.[19] showed that the measurement model could yield estimates for the polarization and ohmic resistances. This approach was applied, for example, to estimate an upper bound for the rate of corrosion of copper in anaerobic water.[20] Orazem et al.[21] used regression of the Voigt measurement model as a means of deconvolution to identify underlying distributions in impedance spectra, and Chen et al.[22] explored the use of the Voigt measurement model to identify resistivity distributions in oxide films. The measurement model was the foundation for the observation that a power-law distribution of resistivity in a film gives rise to constant-phase-element (CPE) behavior.[23,24] More recently, Liao et al.[4] and You et al.[25] demonstrated that, under many conditions, the measurement model my be used to extract the capacitance of electrochemical systems. The ability to extract capacitance by use of the measurement model is important, because most impedance systems show distributed-time-constant behavior that obscures the value of the capacitance.

This document is intended to provide an introductory guide to the measurement model toolbox, a program intended to give to the scientific community the tools we have been using in our group for the past 20 years. With this program, you will be able to identify the stochastic error structure of your measurements, used to weight further regressions. You will be able to determine what part of your measurement is inconsistent with the Kramers–Kronig relations. You will be able to estimate capacitance and ohmic resistance, from which you can identify the characteristic frequency above which the geometry of the electrode may cause frequency dispersion. You will also be able to fit custom models to your data. Some of the concepts which underlie the program are described in Orazem and Tribollet.[6] We hope that this document and the program it describes will help realize the vision that, by providing the tools needed for a more sophisticated analysis of impedance data, we will enable a world in which researchers take full advantage of the information that can be extracted from electrochemical impedance spectroscopy measurements.[26]

William Watson and Mark E. Orazem
Gainesville, Florida

Why an Elephant?

The distorted impedance diagram, shown on the cover of this book, was intended to resemble an elephant, and, in doing so, evoke the lessons of the Buddhist tradition of the blind men and the elephant. The multiple loops resemble the Nyquist plots obtained in some cases for the impedance of corroding systems influenced by formation of surface films. The low-frequency inductive loop was deformed to evoke the image of the elephant's trunk, and the capacitive loops resemble the head and body of the elephant.

The logo for the measurement model program is an elephant, which, as shown in Figure 1, has become a symbol for our group. I chose the elephant logo when I was organizing the 2004 International Symposium on Impedance Spectroscopy. Members of our group have defined a constant-phase elephant, which is deployed in Chapter 14 of Orazem and Tribollet.[6]

Figure 1: Elephants drawn by doctoral student Ming Gao to welcome prospective students to our lab.

As discussed in Orazem and Tribollet,[6] "impedance spectroscopy is an application of a frequency-domain measurement to a complex system that cannot be easily visualized. The quantities measured, e.g., current and potential for electrochemical or elec-

tronic systems and stress and strain for mechanical systems, are macroscopic values that represent the spatial average of individual events. These quantities are influenced by the desired physical properties, such as diffusivity, rate constants, and viscosity, but do not provide a direct measure."

The point is that every experimental measurement we use gives us one view of a system. From the results of this measurement, we can develop a model of the system under investigation. Our confidence in the validity of the model can be improved by performing new observations of the system, often guided by the model we developed. Impedance spectroscopy is a powerful tool, but the elephant reminds us that impedance spectroscopy is not a stand-alone technique.

Chapter 1

Getting Started

The code comes packaged in Microsoft installer which, when executed, will place all the necessary files into the correct locations, assign default file types, and create shortcuts as desired. The program is written in Python and comes in an executable; the executable contains what is essentially a stand-alone Python interpreter. Although this causes the program to be relatively large, it also means that a separate Python installation is not necessary to use the program. An overview of the application of the measurement model toolbox for statistical analysis of impedance data was presented in a series of papers by Agarwal et al.[1–3]

The program can be launched by running `Measurement Model.exe`. A popup will appear displaying copyright information as shown in Figure 1.1. The software is licensed under the GNU General Public License; this can be accessed by clicking the link in the popup or by visiting https://www.gnu.org/licenses/. Press `Start program` or `Enter` to begin the program. Pressing `Cancel` or otherwise closing the alert will terminate the program. There are three main regions to the program layout as shown in Figure 1.2. the title bar (top), the navigation pane (left), and the main window (right).

The title bar serves to indicate the main purpose of the current tab, while clicking an icon in the navigation pane on the left will switch to that particular tab. The current active tab icon is highlighted blue in the navigation pane. General information about a component in a tab can usually be found by hovering over that component for a few seconds and reading the resulting tooltip.

1.1 File Types

The program also makes use of six types of files (see Table 1.1). When the installer runs, the file types will be assigned and receive icons; they can then be opened with the Measurement Model program. The file types will be discussed further as they arise.

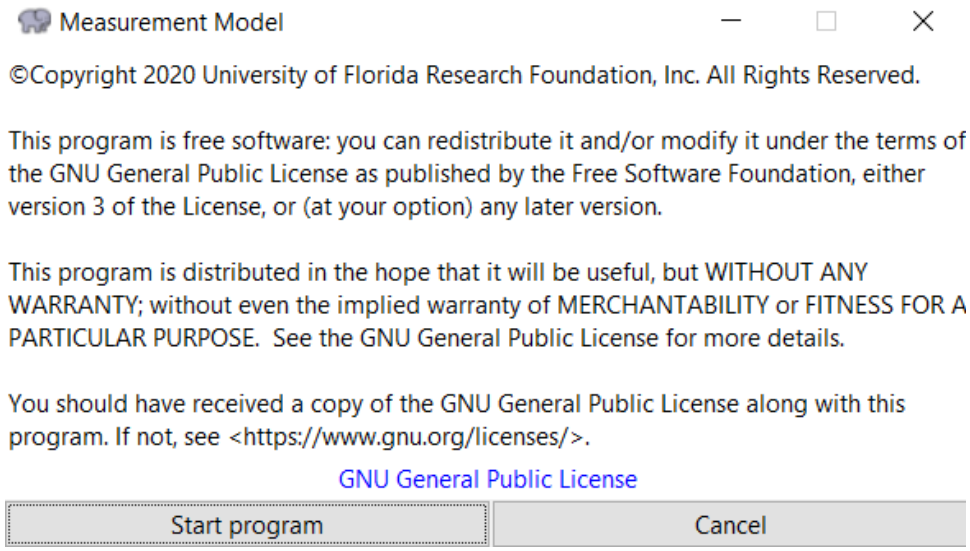

Figure 1.1: Initial copyright popup. This program is distributed under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

Table 1.1: File extensions used by the program.

Extension	Purpose	Image
*.mmfile	Holds impedance data (frequency, real, and imaginary)	
*.mmfitting	Holds information from a previous fit	
*.mmresiduals	Holds residual errors from a fit	
*.mmerrors	A combined set of *mmresiduals* files	
*.mmcustom	Holds a custom fitting	
*.mmformula	Holds a custom formula	

1.2 Program Tabs

The program is broken up into seven tabs (see Table 1.2). The first five tabs represent discrete functions of the program: conversion of data files into the format used by the program, regression of the measurement model, estimation of the standard deviation of the stochastic errors from repeated measurements, identification of a model for the stochastic error structure, and regression by user-defined custom process models. The final two tabs provide access to customization and this document. Each tab is discussed

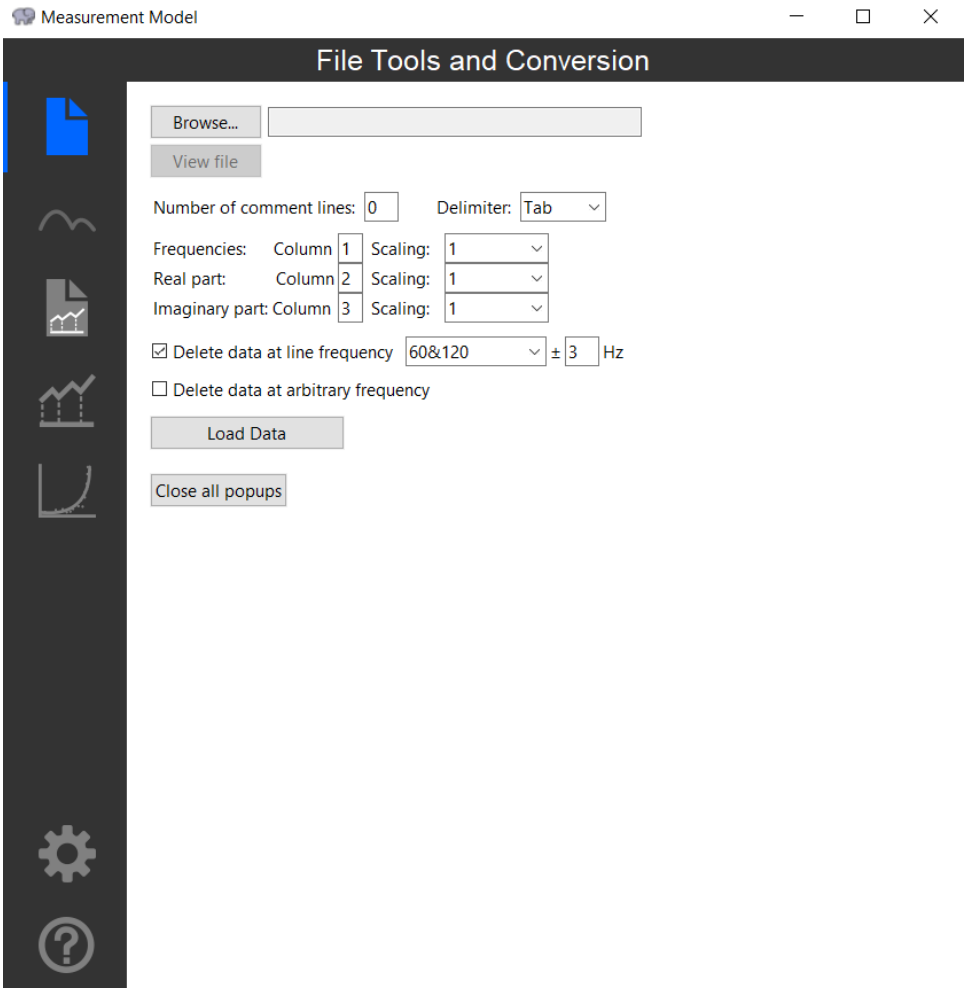

Figure 1.2: The initial program layout. The program opens at the `File Tools and Conversion` tab used to convert data files into the format needed by the program. The other tabs on the left-hand side are described in Table 1.2.

in detail in a subsequent section.

Table 1.2: Program Tabs. The first five tabs represent program functions: data file conversion, measurement model regression, estimation of the standard deviation of the stochastic errors, model identification for the stochastic error structure, and custom model regression. The final two tabs provide access to customization and to this document.

Tab	Image	Purpose
File Tools and Conversion		Convert files to a program-usable format
Measurement Model		Fit data to the measurement model
Error File Preparation		Combine residual error files for analysis
Error Analysis		Fit errors to an error structure
Custom Formula Fitting		Allow for the fitting of custom functions
Settings		Change program preferences
Help and About		Program information and assistance

1.3 Close all Popups

The `Close all popups` button appears at the bottom of every tab (except Help and About). When clicked, it will close all open popups created from all tabs. This can be useful to "clean up" after numerous popups have been opened without closing.

1.4 Sample Files

Sample files included in this distribution are presented in Table 1.4.Please create working directories and copy the files into them. Proposed file directories are provided. These paths should be entered into the Paths page of the Settings tab (see Section 7.2).

Table 1.3: Sample files included in this distribution. Create working directories and copy the files into them. Proposed file directories are provided.

File Type	Link	Proposed Directory
Data	Sample data files	`C:\EIS-EXPT\sample-data`
Formula	Sample custom formulas	`C:\Custom Fitting Models`
Imported code	Sample Python code	`C:\Custom Fitting Models\LUT`

1.4.1 Sample Data Files

The sample data files were collected by Madhav Durbha in Paris, France on May 31, 1995. The potentiostat was a Solartron 1286, and the frequency response analyzer was a Solartron 1250. A matched two-channel Kemo VBF8 48 low-pass Butterworth analog filter was used to reduce the noise level of the input signals to the Solartron 1250. The data were collected from high frequency to low frequency with 12 logarithmically spaced frequencies per decade.[27,28]

The long (1% closure error) auto-integration option of the frequency-response analyzer was used on the current channel. The solution was not deaerated. FraCom software developed in-house at CNRS by H. Takenouti was used to control the experiments and collect the data. The data files presented here were modified to make them compatible with the present measurement model program.

The electrolyte consisted of 0.01 M $K_3Fe(CN)_6$ and 0.01 M $K_4Fe(CN)_6$ in 1 M KCl. The temperature was controlled at 25.0$\pm$0.1°C. The electrode diameter was 5 mm, yielding a surface area of 0.1963 cm^2. A polishing technique was selected for the Pt disk electrode that provided the maximum steady mass-transfer-limited current. The electrode was polished with a 1200 grit emery cloth, washed in deionized water, polished with alumina paste, and subjected to ultrasound cleaning in a 1:1 solution of water and ethyl alcohol. Experiments conducted at 1/4 of the mass-transfer-limited current were replicated. The data used for the present analysis were collected at a rotation speed of 120 rpm.

The data files provided include the first in the series of measurements, represented by the time of 14 s at which the first frequency was completed. The other three measurements were completed 9.6 hours after the beginning of the series of measurements. The first measurement was affected by nonstationary behavior; whereas, the measurements taken after an elapsed time exceeding nine and a half hours could be considered stationary.

1.4.2 Sample Custom Formulas

The sample custom formulas provided are described in Sections 6 and 9.5. These models are intended to provide a beginning to a model library. These include a model with an ohmic resistance in series with a CPE and a faradic impedance that includes a faradaic reaction with the diffusion impedance for a film, a model in which the diffusion impedance is expressed as a Warburg impedance, and a model in which the diffusion impedance is that corresponding to a rotating disk electrode, evaluated using the look-up table file `LUT.py` described in Section 1.4.3.

1.4.3 Sample Python Code

The custom formula regression part of the program is equation-based, meaning that models are input in the form of Python code rather than in the form of built-in electrical

circuit elements. The advantage is that models are not limited to predefined analytic expressions.

The sample Python code provided here enables a look-up table for the convective diffusion impedance response of a rotating disk electrode. The convective-diffusion impedance for the rotating disk is obtained directly from

$$\frac{-1}{\theta_{\mathrm{i}}'(0)} = Z_{(0)} + \frac{Z_{(1)}}{\mathrm{Sc_i}^{1/3}} + \frac{Z_{(2)}}{\mathrm{Sc_i}^{2/3}} + \ldots \tag{1.1}$$

where $-1/\theta_{\mathrm{i}}'(0)$ is the dimensionless diffusion impedance, $Z_{(0)}$, $Z_{(1)}$, and $Z_{(2)}$ are the three complex solutions to the convective diffusion impedance corresponding to three terms in the series expansion for the axial velocity near the disk electrode, and the Schmidt number is expressed as $\mathrm{Sc} = \nu/D_{\mathrm{i}}$, where $\nu = \mu/\rho$ is the kinematic viscosity and D_{i} is the diffusion coefficient for the reacting species. The file `LUT.py` uses an interpolation function and a look-up table (hence, the file name `LUT.py`) to enable evaluation of $Z_{(0)}$, $Z_{(1)}$, and $Z_{(2)}$ at a specified frequency. The mathematical foundation is given in Section 11.3.4 of Orazem and Tribollet[6] and is based on the work of Tribollet and Newman.[29] The provided code is used in one of the custom formulas described in Section 1.4.2.

Chapter 2

File Tools and Conversion

The first step towards analyzing any impedance spectra is to convert the file into a format the program will understand, namely a file with a **.mmfile* extension. These files are simple text files with three columns: the first is frequency, the second is real impedance, and the third is imaginary impedance. Creation of these files is the job of the first tab: File Input and Conversion. Access this tab by clicking the document icon ( ) at the top of the navigation pane. Note that none of the tools in this tab will edit the original file. The input file tab can be seen in Figure 1.2.

2.1 Input

Files of any kind can be loaded by using the `Browse` button. This will bring up the standard operating-system file-open dialog, from whence a file can be chosen. Upon selecting a file, the name of this file will appear in the text bar to the right of the button. The contents of the file can be viewed using the `View file` button. This button is initially disabled and will only be clickable after a file has been chosen with the `Browse` button. The `View file` button will bring up a small popup with line numbers and the raw text of the file.

2.2 Number of Comments

The `Number of comment lines` box indicates the number of lines that the program ignores before reading the impedance data. This number can be set manually; it can be helpful to use the `View file` button mentioned in Section 2.1 to determine on which line number the data start. The program will also attempt to determine the number of comment lines automatically by reading through the file and looking for the first line on which consistently delimited text data appears. This feature can be disabled/enabled in the Settings tab described in Section 7.3.

2.3 Delimiter

The `Delimiter` is the character used to indicate separation between columns. The delimiter can be chosen manually as one of:

- Tab
- Space
- Semicolon ";"
- Colon ":"
- Comma ","
- Pipe "|"

As with the number of comment lines, the program will attempt to determine the delimiter automatically. This action can be disabled in the Settings tab described in Section 7.3.

2.4 Recognized File Types

If the automatic number of comments detection is enabled, the program will also attempt to automatically parse certain common file types. These are listed in Table 2.1. The method to detect the different file types was taken from impedance.py.

Table 2.1: Recognized File Types. The method to detect the different file types was taken from impedance.py.

Company	Extension	Comments
AutoLab	*.txt	
Biologic	*.mpt	
Biologic	*.mpr	This file is binary, not plain text; therefore, a simplified version is parsed and displayed
CH Instruments	*.txt	
Gamry	*.dta	Can detect if experiment was aborted and parse appropriately
Parstat	*.txt	
PowerSuite	*.txt	
VersaStudio	*.par	Will only detect data from the first segment
	*.z	Can detect files with and without metadata

2.5 Columns

The program will look for data separated into three columns: one each for frequency, real impedance, and imaginary impedance. These columns should be separated using the delimiter mentioned in Section 2.3. The column numbers should be entered manually. The `Scaling` is the multiplier used on the input data. The default scaling multiplier is `1` (i.e., the data will be the same as in the input file). Other choices are available from the dropdown menu, and arbitrary values can be entered by typing in the box as well (the value must a real number).

> **Tip!** Some impedance systems report minus the imaginary impedance. It is useful to plot the Nyquist plot to confirm that the impedance you have loaded is in the correct form.

2.6 Deleting Data

Impedance measurements taken near the frequency of the power source can be noisy. Noise from the power source can be superimposed upon the input and output signals of the frequency response analyzer. Even one or two spuriously noisy data points near the line frequency can influence the regression of the impedance spectrum. To avoid this problem, data taken at frequencies close to the first and second harmonic of the power source frequency should be removed from the analysis. The line frequency in the US and Canada is 60 Hz; therefore, impedance measurements taken near 60 Hz and 120 Hz should be deleted from the analysis of data collected in the US or Canada. The line frequency in Europe is 50 Hz; therefore, impedance measurements taken near 50 Hz and 100 Hz should be deleted from the analysis. The `Delete data at line frequency` dropdown provides this functionality. The data at the selected frequency(ies) will be removed in the range specified by the $\pm$ box. For instance, if `50&100` $\pm$ `3` Hz is chosen, then all data from 47-53 and 97-103 Hz would be removed. The default is 60 ± 3 and 120 ± 3 Hz.

Sometimes, data at different frequencies may need to be discarded. The `Delete data at arbitrary frequency` checkbox, when checked, will provide more options for doing so. To the left will be a listbox (initially empty). This will be propagated with frequencies as they are added. The first entry field to the right indicates the main frequency to be deleted, while the second entry field indicates the range around the first frequency to be deleted. For example, entering `42` $\pm$ `5` would delete all data from 37-47 Hz. Click the `Add` button to add this deleted frequency to the listbox (see Figure 2.1).

To remove an item from the listbox (and hence have that frequency range *not* be deleted), simply select the item in the listbox and click the `Remove` button, or right-click the item and select `Remove`.

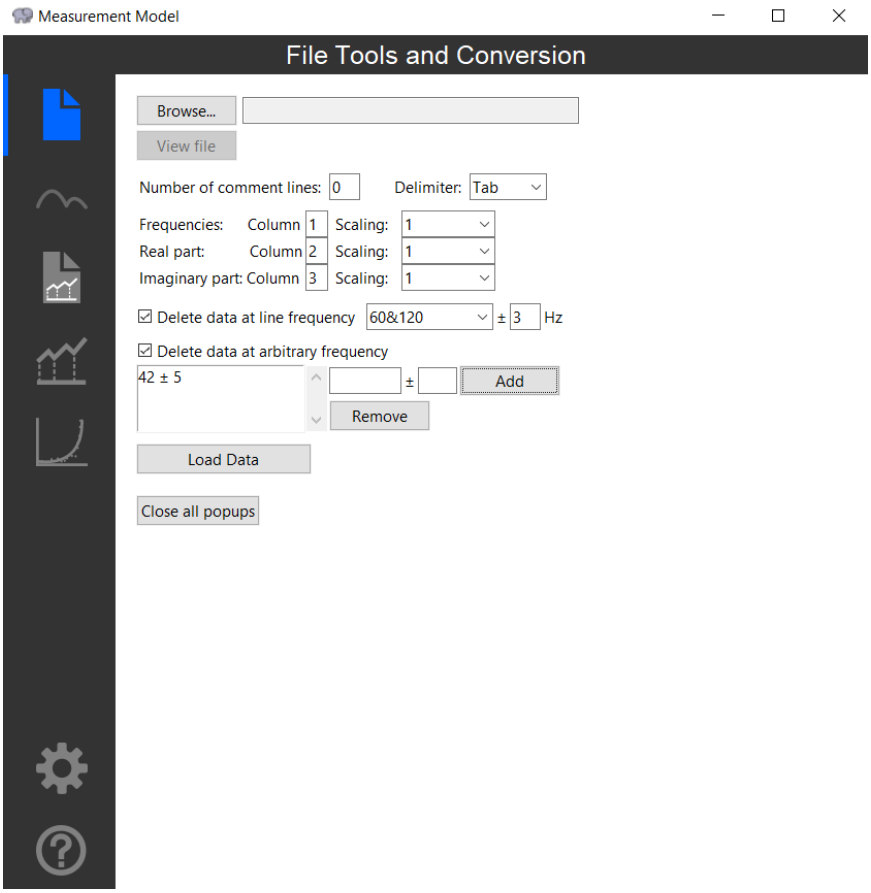

Figure 2.1: Deleting data at arbitrary frequency. The image shows the procedure to delete frequencies of 42 ± 5 Hz. In addition, the check box is selected for deletion of data near the line frequency and its first harmonic, i.e., 60 ± 3 and 120 ± 3 Hz.

Tip! When using auto-integration, measurements taken near the line frequency and its first harmonic (e.g., , 60 ± 3 Hz and 120 ± 3 Hz) can take a long time to converge. To minimize the time required for the impedance measurement, design the experiment to avoid these frequencies rather than making measurements that will subsequently be discarded.

2.7 Loading Data

After all the preceding choices have made, click the `Load data` button to load the data. Once data is loaded, this button will be renamed to `Reload Data`. In order for changes made to the choices to have an effect, the `Reload Data` button must be pressed.

When the data is loaded, a listbox will appear below with four columns as shown in Figure 2.2. Each row of the listbox holds one data point. The first column is the data points "number", the second is the frequency, the third is the real part of the data, and the fourth is the imaginary part. Above this listbox will be the `Number of data`, which will indicate how many data points were loaded.

Below the listbox are three buttons. One is `Save As`, which will save the data shown in the listbox to a **.mmfile*. The second is `Save and Load to Measurement Model`, which will save the data as a **.mmfile* and then open it in the second tab (see Chapter 3). The third is `Nyquist plot`, which will open a popup displaying a Nyquist plot of the data. An example is shown in Figure 2.3. Every plot generated by the program will have the same controls, which are discussed in Section 2.8.

2.8 Graphing Controls

Each plot generated by the program has the same controls, which are shown in the toolbar below a given plot (see Figure 2.3). The leftmost three buttons work similarly to Internet browser control buttons. The leftmost button (with a house) returns the view to its original state. The second button (a back arrow) changes the plot back to the state it was immediately before the current one, which the third button (a forwards arrow) changes the plot to a newer one (if the back button has been clicked).

The fourth button (with two crossing double-headed arrows) allows the plot view to be moved by dragging with the left mouse button. Dragging with the right mouse button will cause the plot to be zoomed. The fifth button (a magnifying glass) allows a rectangular region of the plot to be selected; this area will then be zoomed to be the new view. Use the back or home buttons to zoom out, or the move button by dragging with the right mouse button. The sixth button, used to configure subplots, is generally not needed for the present program. Further information on the configure subplots option can be found by looking up the `matplotlib` GUI control bar. The last button is the save button, which will open a popup to save the current figure as an image file.

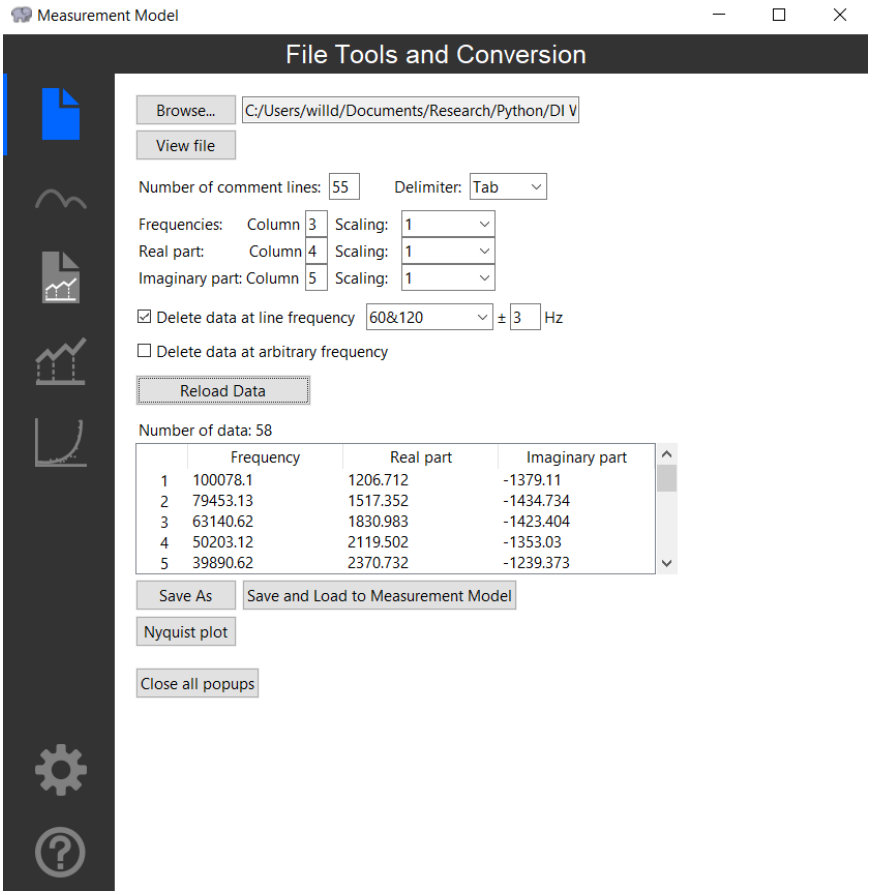

Figure 2.2: Loaded impedance data. The listbox shows a line number, the frequency, and the real and imaginary parts of the impedance. Some impedance systems report minus the imaginary impedance. It is useful to click the `Nyquist plot` button to confirm that the impedance you have loaded is in the correct form.

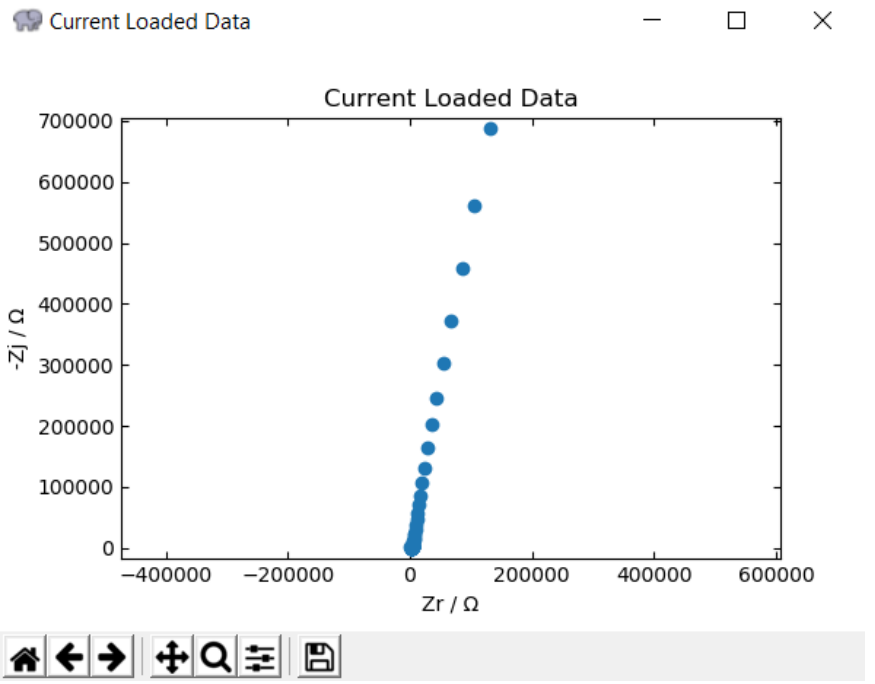

Figure 2.3: A Nyquist plot of current loaded data. This feature is useful to confirm that the impedance you have loaded is in the correct form. The same layout is used for all plots in the program.

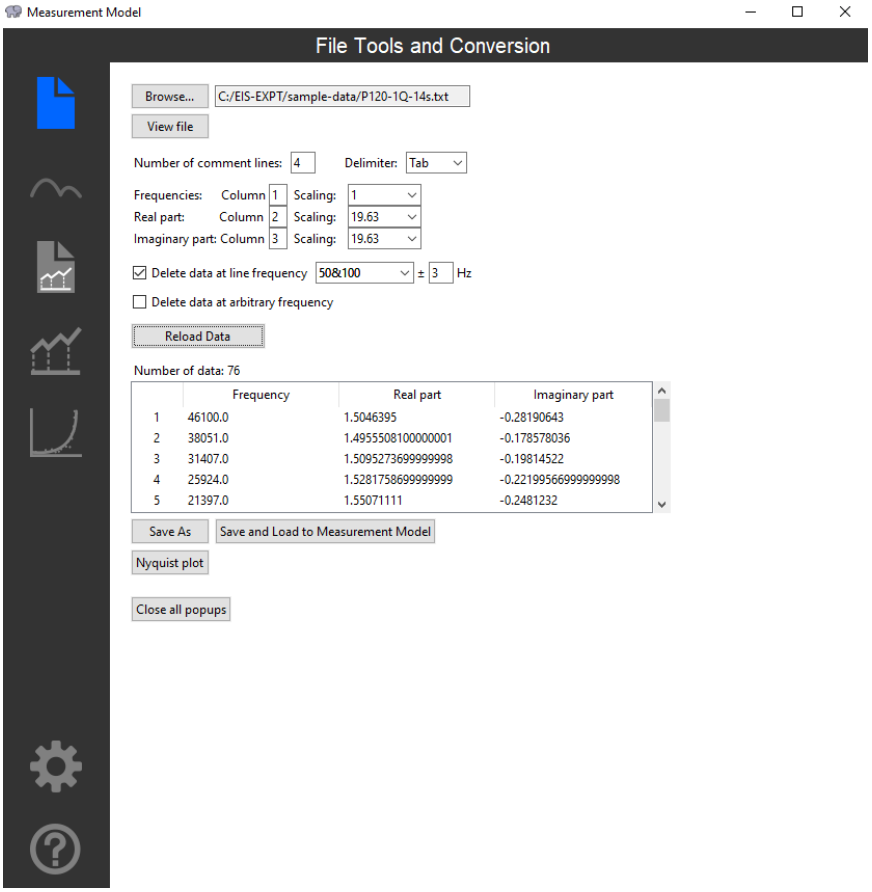

Figure 2.4: Sample file conversion. The number of comments are added manually, and the scaling is given by Rst×Area to yield impedance in units of Ωcm^2. As the data were collected in Europe, data within 3 Hz of the line frequency of 50 Hz and its first harmonic were deleted.

2.9 Application to Sample Files

Browse to the sample file directory and select one of the files. While for many data types, the measurement model program can automatically identify the number of comment lines and the columns corresponding to the impedance data, the program cannot resolve these for the sample data files. Thus, the number of comments must be added manually, and the columns must be selected manually. The default selection is satisfactory. The scaling is given by Rst×Area to yield impedance in units of Ωcm^2. As the data were collected in Europe, we want to delete data within 3 Hz of the line frequency of 50 Hz and its first harmonic. The corresponding panel is shown in Figure 2.4. Save the file and proceed to convert the remaining sample data files. You will need to update the number of comment lines for each new file, but the scaling and column selection is remembered. Reload the file before saving.

If you forget to update the number of comment lines, you will receive an error message “Error 5: There was an error loading or reading file.” Click “OK,” and update the value. At this stage, you will have four files withe the “*.mmfile” extension.

Chapter 3

Measurement Model

After the **.mmfile* has been created using the File Tools and Conversion tab described in Chapter 2, it can be loaded into the Measurement Model tab to be fit to a measurement model consisting of a specified number of Voigt elements. Access this tab by clicking the Nyquist plot icon (◠) at the top of the navigation pane. Note that the tools in this tab will not edit the **.mmfile*. The measurement model tab can be seen in Figure 3.1.

3.1 About the Regression

The measurement model works by regressing a model for impedance to the actual impedance data. This model consists of an ohmic resistance in series with a number of Voigt elements these Voigt elements consist of a resistor in parallel with a capacitor. The general structure for the measurement model to be regressed is[1]

$$\widehat{Z} = R_{\mathrm{e}}\,(+C) + \sum_{\mathrm{i}=1}^{\mathrm{K}} \frac{R_{\mathrm{i}}}{1+\mathrm{j}(2\pi f)R_{\mathrm{i}}C_{\mathrm{i}}} = R_{\mathrm{e}}\,(+C) + \sum_{\mathrm{i}=1}^{\mathrm{K}} \frac{R_{\mathrm{i}}}{1+\mathrm{j}(2\pi f)\tau_{\mathrm{i}}} \tag{3.1}$$

where R_{e} is the ohmic resistance, C is the (optional) capacitance, R_{i} is the resistance of the i-th Voigt element, and τ_{i} is the time constant of the i-th Voigt element (equivalent to $\tau_{\mathrm{i}} = R_{\mathrm{i}}C_{\mathrm{i}}$). A schematic representation of the corresponding circuit is shown in Figure 3.2. The regression procedure uses a Levenberg–Marquardt algorithm to minimize the sum of squared of the objective function

$$\chi^2 = \frac{\left(Z - \widehat{Z}\right)^2}{\sigma^2} \tag{3.2}$$

where Z is the experimental complex impedance, $\widehat{Z}$ is the model value, and σ is the standard deviation of the measured impedance, described in Section 3.7.

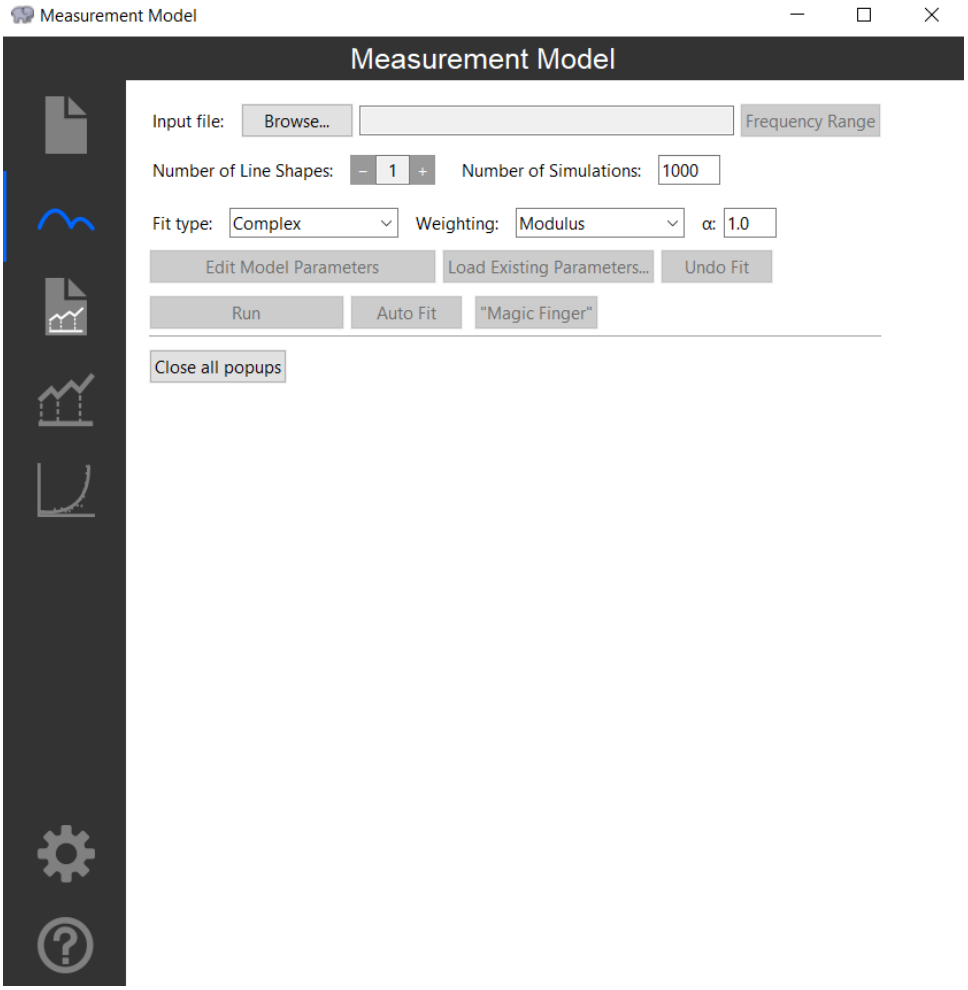

Figure 3.1: The measurement model tab. Other actions will become available after the data file is selected.

3.2 File Input

Files with two types of extensions can be loaded to the measurement model: **.mmfile* and **.mmfitting*. The first holds a single set of impedance measurements and is created via the File Tools and Conversion tab (see Chapter 2), while the second holds a set of fitted parameters and link to an existing **.mmfile*; **.mmfitting* files can be created in the Measurement Model tab after a successful regression is performed. The `Browse` button in the top row can be used to open a dialog and select a file for use. After a file is chosen, its path will appear in the textbox to the right. For **.mmfile*s, the buttons below will be activated upon loading. If a **.mmfitting* file was chosen, the path will be the **.mmfile* used to create the fitting; in addition, fitting results will appear below.

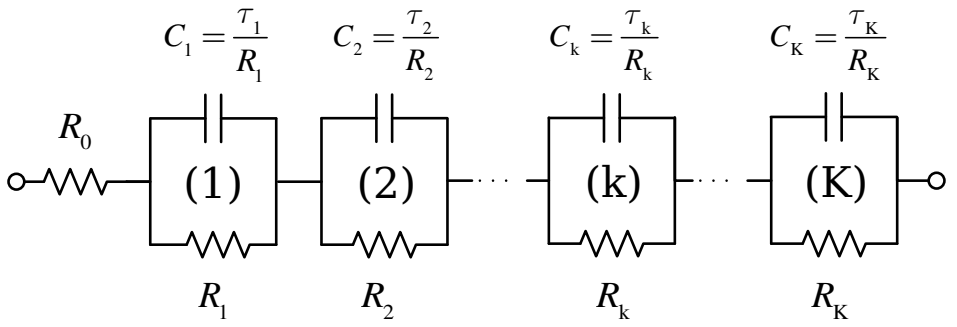

Figure 3.2: A schematic representation of a Voigt circuit used by Agarwal et al.[1–3] as a measurement model. Taken from Liao et al.[4]

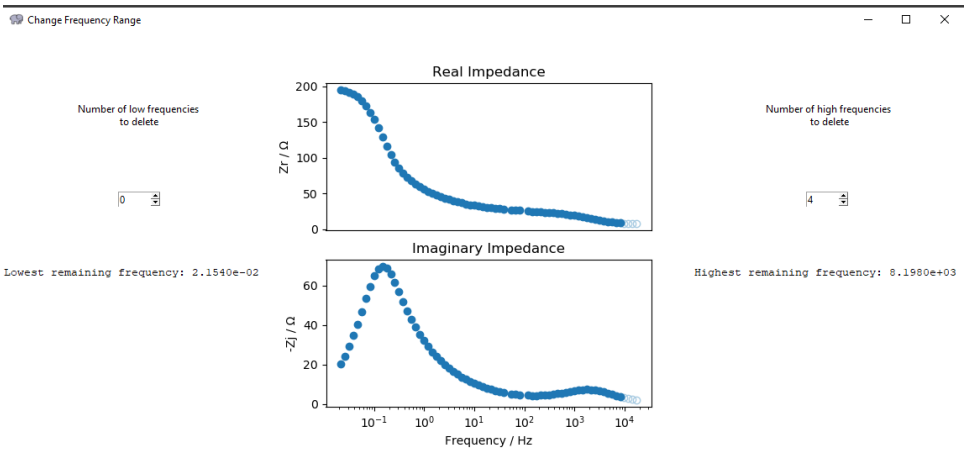

Figure 3.3: Changing the frequencies used for fitting. Note that, as the data shown was collected in Europe, frequencies at 50 and 100 Hz were deleted during the file conversion step. Note that frequencies removed for the present regression are not removed from the file. Removed frequencies are shown in open symbols, and remaining frequencies are shown in closed symbols.

3.3 Changing Frequencies

Depending on the specific data used, it may be necessary to remove high or low frequencies from the fit. This can be done by using the `Frequency Range` button. A popup similar to Figure 3.3 will appear to allow deletion of either the low or high frequencies.

The graph in the middle is used to indicate visually which frequencies have been removed; the upper graph is the real impedance vs frequency, while the lower is minus imaginary impedance vs frequency. The graph shows all frequencies present in the data set. Frequencies in use are shown as filled circles, while removed frequencies are shown as open circles. The arrows on the two spinboxes to either side can be used to change the number of low or high frequencies being removed. The lowest and highest

remaining frequencies (i.e., the ones present after deletion) are shown below the spinboxes; these can be highlighted and copied via keyboard shortcuts. Frequency changes are automatically saved as they are made.

3.4 Number of Voigt Elements

The `Number of Line Shapes` box is used to change the number of Voigt elements being fit. The [-] button on the left is used to decrease the number of shapes (to a minimum of 1) and the [+] button on the right is used to increase the number of shapes. These buttons will not be active until a file is loaded.

> **Tip!** The addition of Voigt elements is sequential. For a new data set of a given type, it is best to start with one Voigt element and to add more elements sequentially until the confidence interval for one or more parameters includes zero.

> **Tip!** The fit of a model consisting of only a resistance and a capacitance is best performed by creating a custom model described in Chapter 6.

3.5 Number of Simulations

The confidence intervals used in the plots are generated through Monte Carlo simulations. The `Number of Simulations` entry field accepts positive integers only and is used to indicate how many Monte Carlo simulations should be performed. A greater number will produce a better/more accurate result, but at the cost of requiring more time. The random number generator used in the simulations is seeded with a constant value of `1234`, which ensures that the exact same fitting with the same number of simulations will result in the same confidence intervals. The default number of simulations is 1000.

3.6 Fit Type

The portion of the data to which the model is fit can be chosen using the `Fit type` dropdown. The options are:

- `Real` - fit to the real part of the data
- `Imaginary` - fit to the imaginary part of the data (cannnot not fit the ohmic resistance)
- `Complex` - fits to both the real and imaginary parts fo the data

The default choice is complex.

3.7 Weighting

Impedance data may span several orders of magnitude. Thus, it is necessary to weight the regression to obtain a decent fit. This can be done by using the `Weighting` dropdown. There are four options for weighting as seen in Table 3.1. The default weighting is `Modulus`.

Table 3.1: Weighting options for regression. When the error structure is unknown, modulus weighting is preferred for potentiostatic modulation; whereas, the no-weighting option is better for galvanostatic modulation under a fixed-amplitude perturbation. Proportional weighting is preferred for synthetic data.

Weighting	Description	Formula
None	No weighting is used	$\sigma_r = \sigma_j = 1$
Modulus	Each frequency is weighted by the modulus of the impedance	$\sigma_r = \sigma_j = \alpha\sqrt{Z_\mathrm{r}^2 + Z_\mathrm{j}^2}$
Proportional	Real and imaginary values are weighted by real and imaginary impedances, respectively	$\sigma_r = \alpha Z_\mathrm{r}$ $\sigma_j = \alpha Z_\mathrm{j}$
Error model	Each frequency is weighted by the experimental error structure (see Chapter 5)	$\sigma_r = \sigma_j = \alpha\left\|Z_\mathrm{j}\right\| + \beta\left\|Z_\mathrm{r} - R_\mathrm{e}\right\| + \gamma\left\|Z\right\|^2 + \delta$

Tip! Modulus weighting provides a good estimate for the error structure of impedance data obtained under potentiostatic modulation or under variable-amplitude galvanostatic modulation. The constant weighting strategy (no weighting) is better for impedance data obtained under fixed-amplitude galvanostatic modulation. Once a refined model for the error structure has been determined, it should be used for subsequent regression analysis.

3.7.1 Alpha

The parameter α is the assumed standard deviation of the fit and is multiplied by the chosen weighting. It is available only for modulus and proportional weighting.

3.7.2 Error Model

Error model weighting uses the error structure (see Chapter 5) to weight the regression. Choosing the Error model option from the dropdown will cause another row to appear with further options, shown in Figure 3.4. Each box can be checked to use it in the error structure. Unchecking a box will cause its entry field to be disabled and will remove

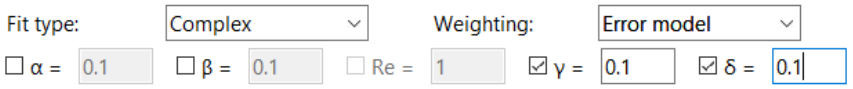

Figure 3.4: Error structure weighting options. The parameters for the error structure model can be obtained using the Error Analysis tab.

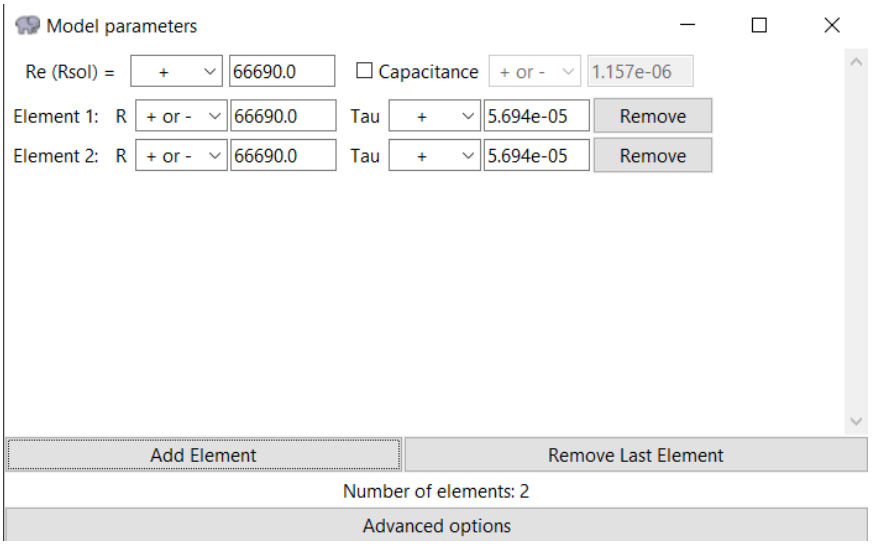

Figure 3.5: Model parameter options. This window is generally not used, but can be useful as a diagnostic for failed regressions caused by time constants that are outside the experimental range. The ohmic resistance may need to be fixed to a value of $R_e = 0$ for data such as those that include a geometric capacitance loop.

it from the error structure. The R_e option is only available if β is chosen as well. If R_e is chosen, the value entered will be used as a constant. In other words, the value entered must be an a priori estimate for the ohmic resistance, as the actual value found through regression may differ from the assumed value. All values must be entered as real numbers.

3.8 Edit Model Parameters

The `Edit Model Parameters` button will open a popup, seen in Figure 3.5, that allows adjusting the current parameters prior to executing a fit. This button is not active until a file has been loaded. Each parameter is listed with a dropdown and an entry field, which are described in Sections 3.8.1 and 3.8.2. When more than one element is present, a `Remove` button will appear next to each element; clicking this will remove that particular element. Checking the box labelled `Capacitance` will add a capacitance in series with the Voigt elements; this is the optional capacitance seen in equation 3.1.

The Add Element and Remove Last Element buttons will add or remove Voigt elements in exactly the same way as the + and - buttons described in Section 3.4. The Number of elements label below these buttons is simply a count of the number of Voigt elements present. The button at the very bottom is Advanced options, which will open another popup, which is described in Section 3.9.

3.8.1 Parameter Bounds

The dropdown by each variable allows limits to be placed on that parameter. There are four options, as seen in Table 3.2. Negative and no bounds (- and + or -) are available for the resistances and capacitance, but not for time constants.

Table 3.2: Parameter bounds. Time constants are assumed to have a positive value, but resistances can be either positive or negative. As time constants are always positive, a negative resistance contributes to fitting an inductive feature in the impedance.

Option	Meaning
+	Force the parameter to be positive
-	Force the parameter to be negative
+ or -	No bounds on the parameter
fixed	Fix the parameter (i.e., do not fit)

3.8.2 Parameter Initial Guesses

Each parameter also has an entry field next to it. The value in this field is used as the initial guess for the regression. Default initial guesses are calculated before any regression is performed based on the data being used. The formulas for these guesses are shown in Table 3.3. After fittings have been performed, the initial guess formulas are updated, as shown in Table 3.4.

Tip! The value of τ_i is constrained by the frequency range of the measurement. For an upper frequency limit of 10 kHz, the initial guess for τ_k should not be smaller than 10^{-6}s.

Tip! The automated guess of parameters for added line shapes is based on a logarithmic average of the time constants. This works well for capacitive systems, but not so well for systems showing inductive behavior. Manual selection of new guesses is necessary to model capacitive systems, best performed using the magic finger described in Section 3.14.

Tip! Remember that $\tau_i = 1/(2\pi f_i)$. To address a poor fit at low frequency, a large time constant is required. For an inductive loop at low frequencies,

Table 3.3: Default initial guesses. The formulas presented here are based on values of the impedance. The resistance is assumed to be the mean resistance, and the time constant is chosen to be between the highest and lowest time constant associated with the respective frequencies.

Parameter	Formula
R	$\left(\max\left(Z_{\mathrm{r}}\right)+\min\left(Z_{\mathrm{r}}\right)\right)/2$
τ	$\dfrac{\left(\log_{10}\left(\dfrac{\max(f)}{\min(f)}\right)\right)}{\left(\max(f)-\min(f)\right)}$
C	$-1/\left(2\pi\left(f\right)Z_{\mathrm{j}}\left(\min\left(f\right)\right)\right)$

Table 3.4: Updated initial guesses for n Voigt elements. The formula for estimating initial guesses for updated Voigt elements is based on the current values of parameters. When this method is insufficient, use the magic finger described in Section 3.14.

Parameter	Formula
R	$\left(\sum_{i=1}^{n} R_i\right)/n$
τ	$10^{\left(\sum_{i=1}^{n}\log_{10}(\tau_i)\right)/n}$

select a large time constant and an negative resistor value. This refinement of initial guesses may be performed manually using Figure 3.5, but is best performed using the magic finger described in Section 3.14.

Tip! If the program fails to converge when a lineshape is added, try using a smaller guessed value for R_{i}.

3.9 Advanced Options and Multistart

The `Advanced options` button is present in the Edit Model Parameters popup. When clicked, it will open a separate popup window seen in Figure 3.6 with further choices for each parameter. The bar on the left allows a particular parameter to be chosen; when chosen, the parameter will be highlighted on the left, and its name will appear at the top of window to the right.

Each parameter has a checkbox labelled `Multistart`. Checking this will activate multistart for that parameter and will enable the other fields present. Multistart works

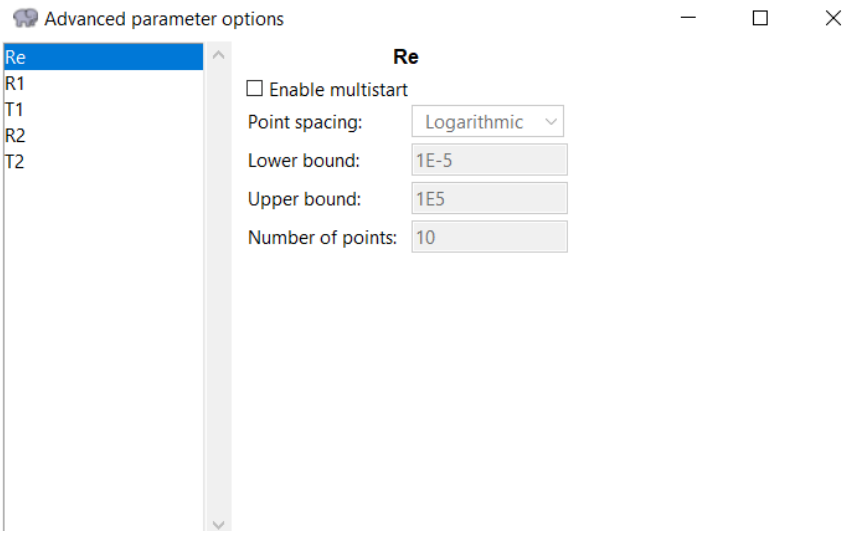

Figure 3.6: Advanced parameter options. The purpose of this window is to allow a automated adjustment of parameters to facilitate a search for a global minimum.

by causing fits to be done at multiple different initial guesses; how these guesses are chosen can be specified in the Edit Model Parameters popup. `Point spacing` has four options:

- `Linear` spaces the points linearly.
- `Logarithmic` spaces the points logarithmically (therefore the bounds must be on one side of zero only).
- `Random` chooses random points between the bounds.
- `Custom` allows for specific guesses to be input.

`Lower bound` and `Upper bound` are the limits between which the guesses will be drawn. `Number of points` is the number of initial guesses chosen. The initial guess in the Edit Model Parameters popup will be tried in addition, thus, the total number of trials for a parameter will be `Number of points`+1. Note that if multiple parameters have multistart activated, each parameter combination will be tried. This means that multistarting multiple parameters will be exponentially slower. If `Custom` spacing is chosen, the other options will disappear and an entry field will appear as in Figure 3.7.

3.10 Load Existing Parameters

Previously fit parameters can be loaded by clicking the `Load Existing Parameters...` button; these parameters should be located in a **.mmfitting* file. Unlike loading a **.mmfitting* file with the `Browse` button, the `Load Existing Parameters...` button will only

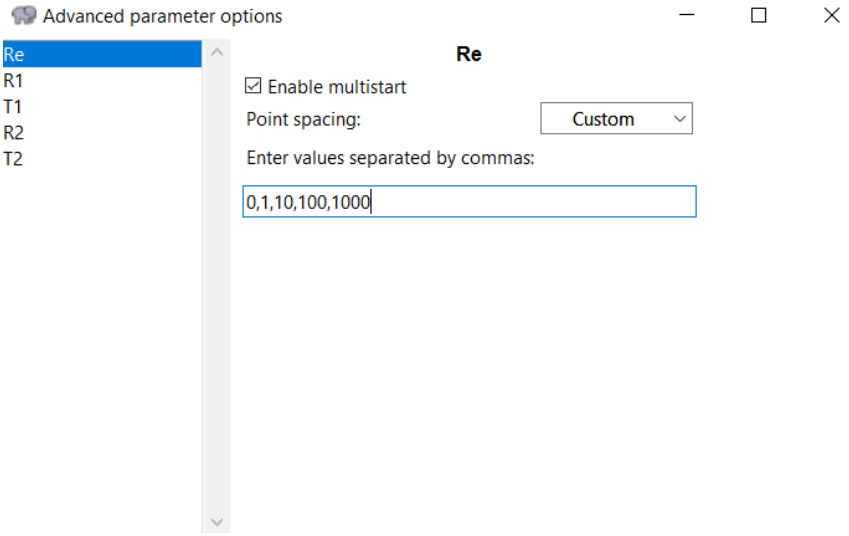

Figure 3.7: Custom multistart guesses. The purpose of this window is to allow a manual adjustment of parameters to facilitate a search for a global minimum.

load the parameters as initial guesses. It will not change other settings, and it will not load a new **.mmfile*.

3.11 Undo Fit

When a bad fit is performed, it is necessary to return to the previous fit. Regression recovery can be achieved with the `Undo Fit` button. The `Undo Fit` button will only be active once at least two fittings have been performed. This will reset all fit parameters to their previous states, as well as changing back all fitting settings, such as `Weighting` and `Fit Type`.

> **Warning!** Once a fit is undone, it cannot be recovered automatically (i.e., the fitting will have to re-performed), so caution is warranted when using this feature.

3.12 Run

The fitting is performed by using the `Run` button. This will perform the regression discussed in Section 3.1 using the settings mentioned throughout this chapter. The button will only become active once a file is loaded. When a regression begins, all other buttons and entry fields will be disabled, including those in popups. These will be re-enabled once the regression is completed.

If a fitting is performed with multistart enabled on a computer with multiple cores, new processes may be spawned to parallelize the computations. This parallelization will occur if the number of parameters multiplied by the number of combinations is greater than 1000. Multiple processes will allow CPU-bound computations to be done in parallel, but there is substantial overhead in creating these processes. The objective of this part of the code is to allow parallel computation when this tradeoff is profitable. The number of processes spawned will be equal to the number of logical cores available.

3.12.1 Progress Bars

The progress of the fit is indicated by a progress bar. If multistart is not chosen, the completion percentage cannot be estimated, and an "indeterminate" progress bar is shown, as in Figure 3.8. If multistart is chosen, the progress bar will show how many

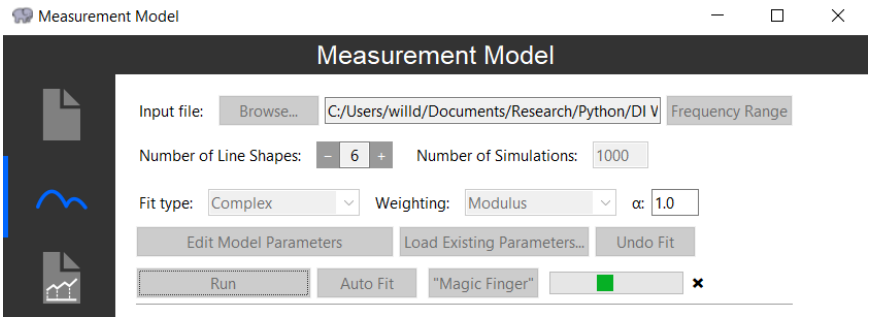

Figure 3.8: Indeterminate progress. A percent completion is not shown, and the motion of the bar indicates that the regression is proceeding.

multistart combinations have been tried; it will also provide information as to when processes are being created and when Monte Carlo simulations are occurring. The resulting progress bar is shown in Figure 3.9. On Windows 7 and above the state of the

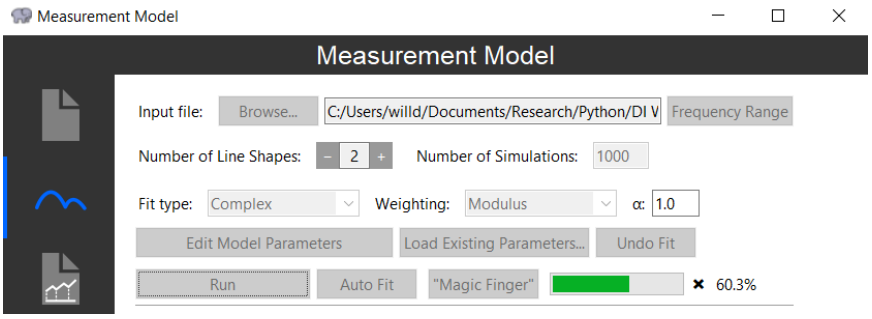

Figure 3.9: Determinate progress. A percent completion is shown for multi-start regressions, and the motion of the bar indicates that the regression is proceeding.

progress bar will be mirrored in the programs taskbar icon.

3.12.2 Cancelling

Once a fitting is started, it can be cancelled by clicking the `X` next to the progress bar. This will kill the fitting thread, as well as any spawned processes, and will leave the current fit in place. All buttons will be reactivated. A "cancel" also occurs when the overall program is closed.

3.12.3 Finishing

When the fitting is finished, a number of things can occur:

- Nothing, if the fitting was cancelled.
- An alert on failure.
- The results of the regression appear.

The regression results are discussed in Section 3.15 and subsequent sections.

3.13 Auto Fit

Clicking the `Auto Fit` button will bring up a popup shown in Figure 3.10, with settings that allow the program to attempt to automatically fit the loaded **.mmfile*. Most of the options are exactly as in the standard Measurement Model tab, including `Number of Simulations` (see Section 3.5), `Weighting` (see Section 3.7), and `Fit type` (see Section 3.6).

3.13.1 Max Voigt Elements

The `Max Voigt Elements` entry takes positive integers only; if the program can do so, it will fit up to this number of line shapes, inclusive.

3.13.2 Fix Ohmic Resistance

The value of R_e can be fixed (i.e., not fit) by using the `Fix` R_e checkbox; this will enable the textbox to the right, where the value of R_e can be set.

3.13.3 Run

Upon clicking `Run`, the program will attempt to automatically fit the data up to the specified number of line shapes, using the specified options, by sequentially stepping and fitting. The program uses the initial guess heuristics described in Table 3.4 for each new step. While the fitting is running, other options will be disabled (in the same manner as when a normal fit is run). A fitting is considered to fail if a fit cannot be found,

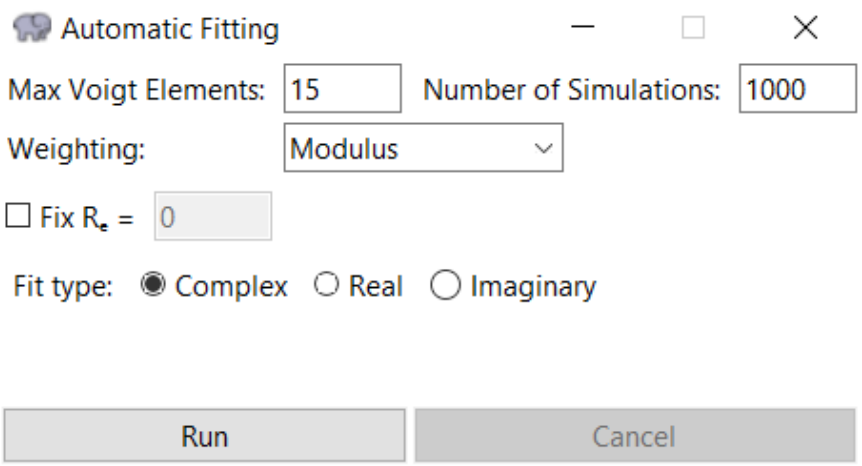

Figure 3.10: The auto fit window. This window appears when the `Auto Fit` button is clicked. Auto fit is an alternative to the semi-automated sequential addition of Voigt elements. When auto fit fails to yield a solution, a successful regression may often be achieved by use of the magic finger described in Section 3.14.

if the confidence intervals on the parameters cannot be estimated, or if the confidence intervals are greater than 100% of the parameter value. Once a fitting fails, the program will attempt a 10-point multistart on the last fit's τ value (from $10^{-5} - 10^{5}$). If this fails and fewer than 5 Voigt elements have been fit, it will attempt to increase the number of line shapes; otherwise, the fitting will end and the results will be displayed in the same way as if a normal fitting had been done.

If `Run` is clicked after a fit has already been performed and results are displayed, a confirmation box will appear. Clicking `OK` will cause the autofit to run; clicking `Cancel` will return to the autofit window. If a regression is already running from the main window, the program will not let you run an autofit as well.

During the fit, a textbox will appear in the autofit window showing which element the program is currently trying. For example

```
Trying 8 Voigt elements with multistart fit...
```

When auto fit fails to yield a solution, a successful regression may often be achieved by use of the magic finger described in Section 3.14.

3.13.4 Cancel

Once a fitting is running, the `Cancel` button will activate. If clicked, it will stop the fit completely, and none of the fitted parameters will be displayed as results.

Tip! If an autofit looks poor when `Plot` (see Section 3.19) is clicked, try clicking the main window's `Run` button to attempt a fit using autofit's pa-

rameters as initial guesses.

3.14 The "Magic Finger"

The "Magic Finger" is a graphical option to choose initial guesses for a new parameter. Clicking the button will open a popup (see Figure 3.11) showing a Nyquist plot with the data points and the current fit, if any. The plot operates in the same manner as

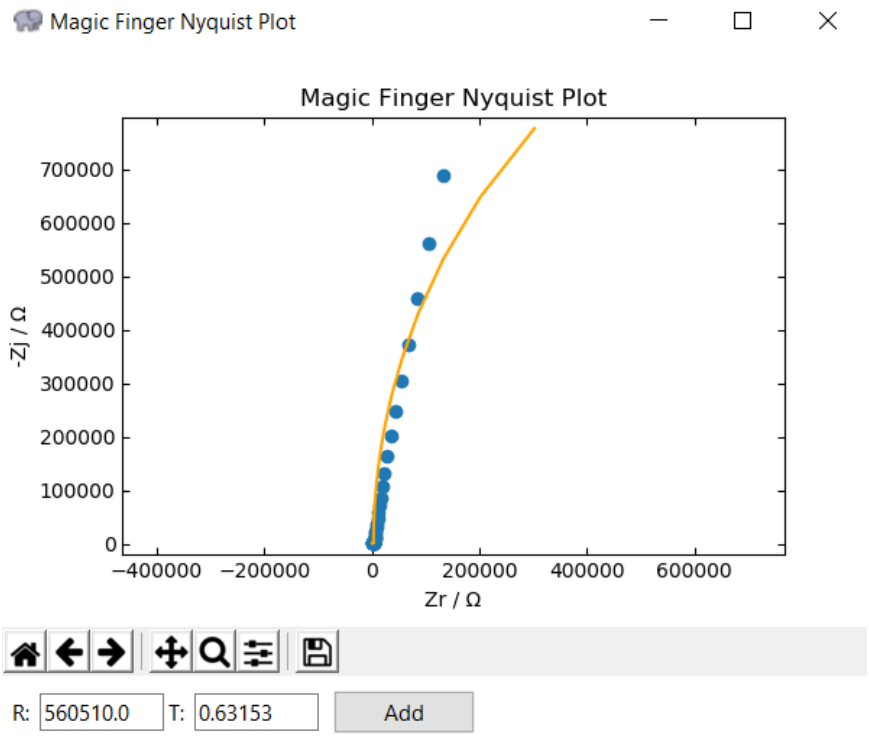

Figure 3.11: The "Magic Finger" window. For the present data set, it will be useful to click on the low-frequency data to add parameters with a large time constant. Lack of fit in the high-frequency range may be seen by using the zoom (magnifying glass) feature of the plot box.

other plots (see Section 2.8). Once a point is clicked, the initial guess corresponding to that point will be shown in the text boxes at the bottom of the window; they can be modified by typing in the boxes. Click `Add` to add the new parameter. If the guess is not a real number, an error will appear. Once added, the text boxes will display zeros.

The best practice is to add parameters corresponding to regions in which the model provides an inadequate fit to the data. In many cases, these frequency ranges may be seen by using the zoom (magnifying glass) feature of the plot box (see Section 2.8).

3.15 Regression Results

After a fit is successfully performed, the results will appear in the bottom half of the tab as shown in Figure 3.12 for a 10-Voigt-element fit for the P120-1Q-34472s file. The

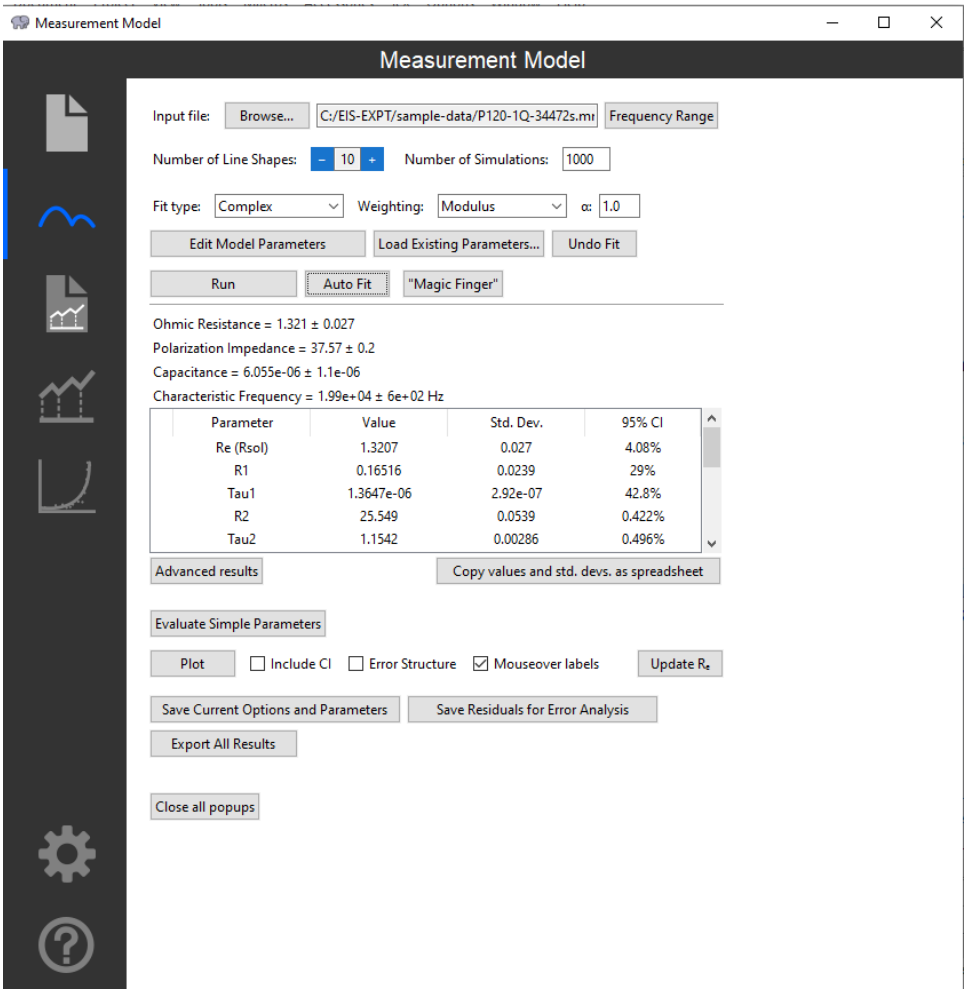

Figure 3.12: Fitting results for 10 Voigt elements for P120-1Q-34472s. The results are shown in a textbox and can be plotted, exported as a *.txt file, and copied for pasting in a spreadsheet.

first results listed are the global system parameters, including the ohmic resistance R_e, the polarization resistance $R_p = Z(0) - R_e$, capacitance $C = (\sum_k R_k/\tau_k)^{-1}$, and characteristic frequency $f_c = 1/2\pi R_e C$. The results shown in Figure 3.12 are:

Ohmic resistance	1.321 ± 0.027	
Polarization Resistance	37.57 ± 0.20	
Overall Capacitance	$6.055 \times 10^{-6} \pm 1.1 \times 10^{-6}$	
Characteristic Frequency	$1.99 \times 10^4 \pm 6.0 \times 10^2$	Hz

Units are not shown as these depend on the units of the data set. If your data are in units of Ω, the ohmic and polarization resistances with have units of Ω, and the capacitance will have units of F. Conversely, if your data are in units of Ωcm^2, the ohmic and polarization resistances will have units of Ωcm^2, and the capacitance will have units of F/cm^2.

Below system parameters will appear a listbox with each fitted parameter name, estimated value, estimated standard deviations, and confidence intervals. The 95% confidence interval is calculated as $200\sigma/p$, where σ is the estimated standard deviation and p is the parameter value. When the confidence interval given in the fourth column is greater than 100 percent, the parameter can have no statistical significance, and the number of parameters should be reduced. Typically, the number of parameters is increased incrementally up to the point where the confidence interval for one or more parameters includes zero.

If a parameter has a confidence interval greater than 100% (or a confidence interval of `nan`), it will be highlighted yellow in the listbox and a red alert triangle will appear to the top right of the listbox, as shown Figure 3.13.

> **Tip!** The success of a non-linear regression is sensitive to the initial guess for parameters. If the quality of the fit is poor, but the confidence intervals indicate that a maximum number of parameters has been achieved, consider a) trying a different initial guess, b) trying a multistart fit, or b) checking the data for consistency with the Kramers–Kronig relations.

3.16 Advanced Results

Further information about the fitting can be found under the `Advanced results` button. The information included in the popup is:

- The file name that was fit
- The amount of data that were fit
- The number of parameters fit, i.e., $N_{\text{param}} = 2 * N_{\text{RT}} + 1$
- The parameters and standard deviations, with more digits
- The zero freqency impedance
- The polarization impedance, with more digits
- The capacitance, with more digits
- A correlation matrix of every parameter
- The χ^2 statistic

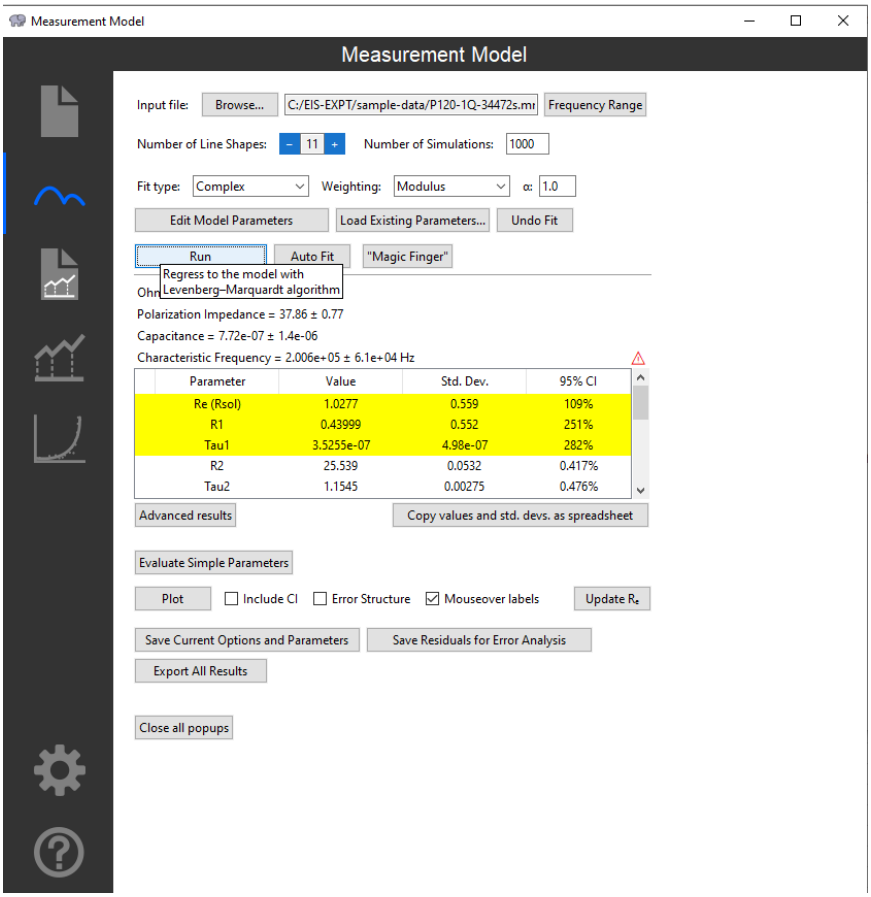

Figure 3.13: A bad fit for the regression of 11 Voigt elements to the data used in Figure 3.12. The ohmic resistance was obtained with a value of 1.0277 with a standard deviation of 0.559. The 2σ confidence interval for the parameter p includes zero, indicated by a value of $95\%CI = 200\sigma_p/p \geq 100$. This condition is indicated by a red alert triangle will appear to the top right of the listbox and a yellow highlight of the offending parameter.

- The χ^2 statistic divided by the degrees of freedom
- The Akaike Information Criterion (AIC)

For example, the results of a 2-line-shape fit of P120-1Q-34472s is shown in Figure 3.14.

Tip! The weighted χ^2 statistic divided by the degrees of freedom, i.e., χ^2/ν, is one of the more important values reported in the Advanced Results page. The value of χ^2/ν should be on the order of unity for an excellent fit, but depends on the weighting strategy used.

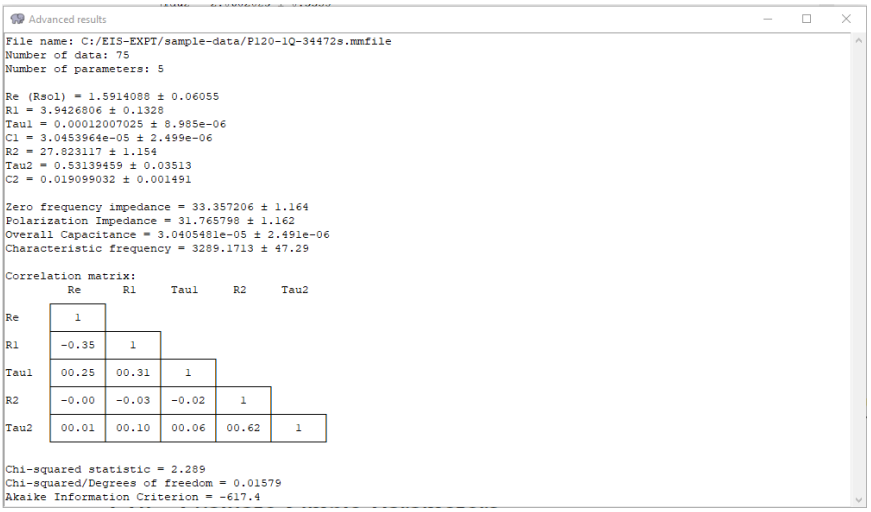

Figure 3.14: Advanced results for a 2-line-shape fit of P120-1Q-34472s. One of the more important values reported in the page is the weighted χ^2 statistic divided by the degrees of freedom, i.e., χ^2/ν. This value should be on the order of unity for an excellent fit, but the value is sensitive to the weighting used.

3.17 Copy Values and Std. Devs. as Spreadsheet

To export the results of a fit in a manner that can be pasted easily into a spreadsheet program, use the `Copy values and std. devs. as spreadsheet` button. This will sort all parameter pairs by ascending time constant, and will also include the file name, capacitances for each parameter pair, the zero-frequency resistance, the polarization resistance, and the overall capacitance.

3.18 Evaluate Simple Parameters

The `Evaluate Simple Parameters` button uses a zero=finding routine to identify the maxima and minima of the imaginary part of the impedance model in order to identify characteristic frequencies, time constants, and capacitances. This can cause a delay of a second or two between when the button is clicked and when the window is ready.

To the right will be a listbox containing all the identified maxima and minima, along with their corresponding characteristic frequencies, time constants, and capacitances calculated according to

$$C = \frac{1}{(2\pi f_c R_p)} \tag{3.3}$$

The parameters R_0 and R_p are calculated and displayed below the listbox as shown in Figure 3.15. To the left of the window is a graph with three tabs above. Clicking these tabs will change the currently displayed graph. `Nyquist` will show a Nyquist plot (the

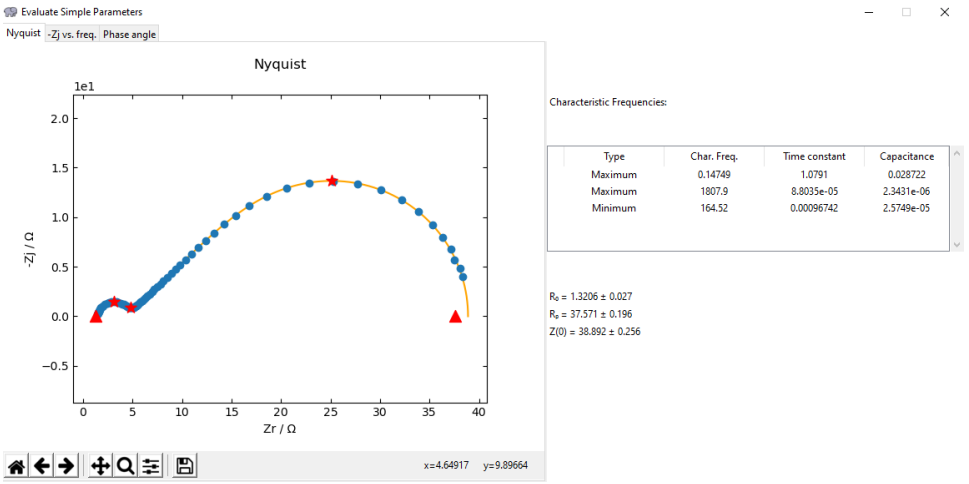

Figure 3.15: Evaluation of characteristic frequencies. The extrapolation is based on a fit of a model that is known to satisfy the Kramers–Kronig relations. Nevertheless, extrapolation is extremely dangerous. Whenever possible, it is better to have experimental data at the characteristic frequencies.

default), `-Zj vs. freq.` will show a graph of the imaginary impedance as a function of frequency, and `Phase angle` will show a graph of the phase angle as a function of frequency.

The red stars in each graph correspond to a maxima or minima shown in the listbox (for the phase angle graph, the stars represent inflection points, not maxima or minima). Hovering over a point will show its frequency, Z_j (for Nyquist and imaginary), Z_r (for Nyquist), and phase angle (for phase angle). The red triangles in the Nyquist plot represent the asymptotic behavior of the impedance.

> **Tip!** Spurious characteristic frequencies will be identified unless the model provides a good fit to the data.
>
> **Warning!** The extrapolation shown in Figures 3.15 is based on a fit to the data of a model that is known to satisfy the Kramers–Kronig relations. Nevertheless, extrapolation is extremely dangerous. Whenever possible, it is better to have experimental data at the characteristic frequencies.
>
> **Warning!** Note that the parameters obtained by the simple analysis do not substitute for development and regression of a process model based on physical and chemical arguments.

3.19 Plot

The `Plot` button will open a full-screen window containing twelve plots as shown in Figure 3.16. The plots include:

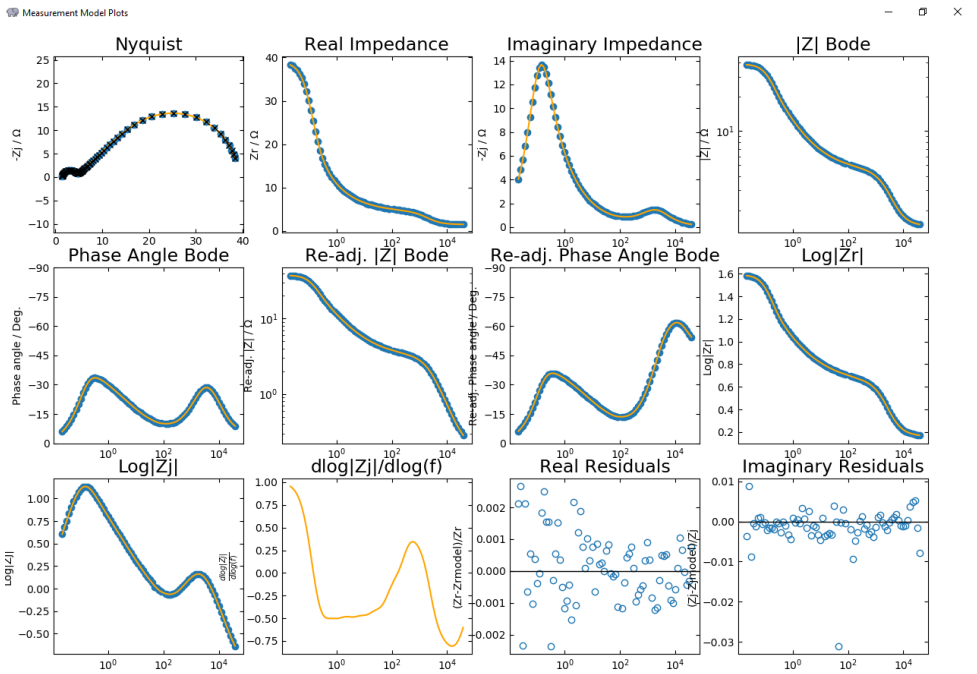

Figure 3.16: Plots of a 10-line-shape fit. The plots include (left to right) a Nyquist plot, real and imaginary impedance as functions of frequency, magnitude as a function of frequency, phase angle (in degrees) as a function of frequency, ohmic-resistance-corrected magnitude as a function of frequency, ohmic-resistance-corrected phase angle as a function of frequency, logarithm of the real and imaginary impedance as functions of frequency, derivative of the logarithm of the imaginary impedance figure, and normalized residual errors. Some of these plots are discussed in Orazem et al.[5] and Orazem and Tribollet.[6]

- Nyquist
- Real impedance as a function of frequency
- Imaginary impedance as a function of frequency
- Modulus, i.e., $\left(\sqrt{Z_r^2 + Z_j^2}\right)$
- Phase angle, i.e., $\left(\arctan\left(Z_j / Z_r\right)\right)$
- Ohmic-resistance-corrected modulus, i.e., $\left(\sqrt{\left(Z_r - R_e\right)^2 + Z_j^2}\right)$
- Ohmic-resistance-corrected phase angle, i.e., $\left(\arctan\left(Z_j / \left(Z_r - R_e\right)\right)\right)$
- Base-10 logarithm of real impedance ($\log_{10}|Z_r|$) as a function of frequency
- Base-10 logarithm of imaginary impedance ($\log_{10}|Z_j|$) as a function of frequency

- Derivative of base-10 logarithm of imaginary impedance with respect to base-10 logarithm of frequency, i.e., $\left(\frac{\partial \log_{10}|Z_j|}{\partial \log_{10} f}\right)$
- Real normalized residuals $\left(\left(Z_r - \widehat{Z}_r\right)/Z_r\right)$
- Imaginary normalized residuals $\left(\left(Z_j - \widehat{Z}_j\right)/Z_j\right)$

Some of these plots are discussed in Orazem et al.[5] and Orazem and Tribollet.[6] Clicking on a plot will open a larger popup version of that plot, which can be operated on using the controls described in Section 2.8. The x-axis labels do not appear for the grouped plots in order to save space; however, they are present in the larger popup version. Note that the plots do not auto-update when a new fit is performed, thus, the `Plot` button must be clicked again.

3.19.1 Include Confidence Interval

If the `Include Confidence Interval` checkbox is checked when the `Plot` is clicked, the window will include confidence intervals around every plot except the ohmic-resistance-corrected Bode plots and the derivative plot, as shown in Figure 3.17. The

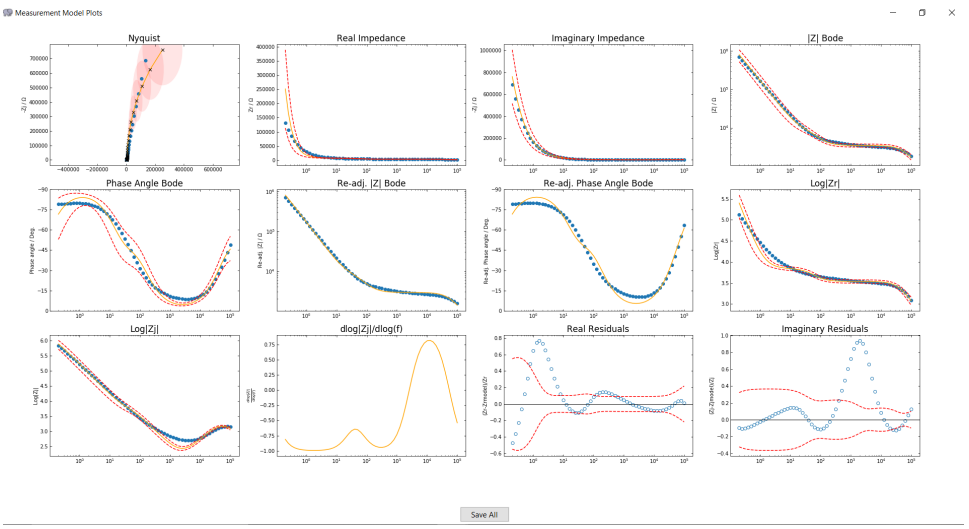

Figure 3.17: Plots with confidence intervals for a fit of three Voigt elements to data for a Pt electrode in deionized water. For a good fit, the confidence intervals are most visible in the residual error plots.

Nyquist plot will include confidence ellipses with a width that indicates the deviation in the Z_r direction and height that indicates the deviation in the Z_j direction.

3.19.2 Mouseover Labels

When checked (the default), the `Mouseover labels` checkbox will cause small labels to appear next to data points when the mouse is hovered over them. This occurs on the larger popup plots, but does not occur on the main grouped plot window. These labels include the frequencies and "y-axis" data; for the Nyquist plot and real and imaginary impedance plots, both Z_r and Z_j are shown.

3.19.3 Save All

The `Save All` button located at the bottom of the plots window in Figures 3.16 and 3.17 can be used to save all of the plots simultaneously. A prompt will appear asking for the directory in which they will be saved. Plots will be saved as PNG files; the name format is *file name* + - + *type of fit and number of Voigt elements* + *_plot type*. The x-axis labels will appear on the saved plots.

3.20 Update Ohmic Resistance

When performing an imaginary fit, it may be necessary to manually adjust the ohmic resistance. The `Update Re` button will open a popup, shown in Figure 3.18. To the left is a graph of the real residuals that will be automatically updated as the R_e is updated. The ohmic resistance can be entered manually in the box to the right. Alternatively,

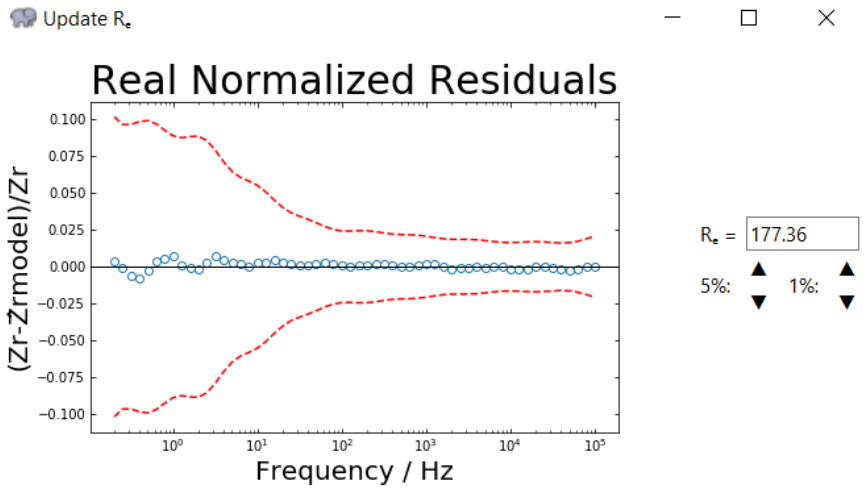

Figure 3.18: The ohmic resistance updating window. A regression to the imaginary part of the impedance cannot provide a value for the ohmic resistance. Thus, the ohmic resistance may be used as an adjustable parameter in determining whether the real part of the impedance falls within the 95.4% confidence interval for the model.

the update and down arrows below can be used: the left arrows will change the ohmic

resistance by 5%, and the right arrows will change the ohmic resistance by 1%. There will be a confirmation alert prior to updating the ohmic resistance, as updating will cause the current fit and confidence interval for R_e to be lost.

3.21 Save Current Options and Parameters

The `Save Current Options and Parameters` button will create a new *.*mmfitting* file. This file includes all currently fitted parameters, the current fitting options (number of Voigt elements, number of simulations, fit type, weighting, parameter constraints, and frequency changes (if any), and a link to the current file, i.e., *.*mmfile*. The entire current fit can then be reloaded in the future (see Section 3.2) or just the parameters can be loaded (see Section 3.10).

3.22 Save Residuals for Error Analysis

The `Save Residuals for Error Analysis` button will create a new *.*mmresiduals* file consisting of the current frequencies, data, fitted ohmic resistance, and real and imaginary residuals. This file can be used to produce a *.*mmerrors* file that is needed for the Error File Preparation tab described in Chapter 4. Subsequently, an error structure model can be obtained in the Error Analysis tab described in Chapter 5.

3.23 Export All Results

The `Export All Results` button will create a *.txt* file containing all information about the current fitting results, including the frequencies, data, weighting, and model values; the fitted parameters (including capacitances) and their confidence intervals; and the chi-squared value. The resulting file can be readily imported into a graphing program such as Excel® or Origin® to create publication-ready graphs.

3.24 Application to Sample Data

The steps taken in this analysis are identification of stochastic error structure, assessment of consistency with the Kramers–Kronig relations, and extraction of physical properties. The regression of process models or custom formulas is presented in Section 6.9.

3.24.1 Identification of Stochastic Error Structure

The first step in the measurement model analysis is to assess the error structure from repeated impedance measurements. To do this, use the three data sets completed 9.6 hours after the beginning of the series of measurements. Begin with the file P120-1Q-32135s.mmfile. Remove the first measured frequency as it often is affected by the transient change from steady state to the sinusoidal condition (see Figure 3.19).

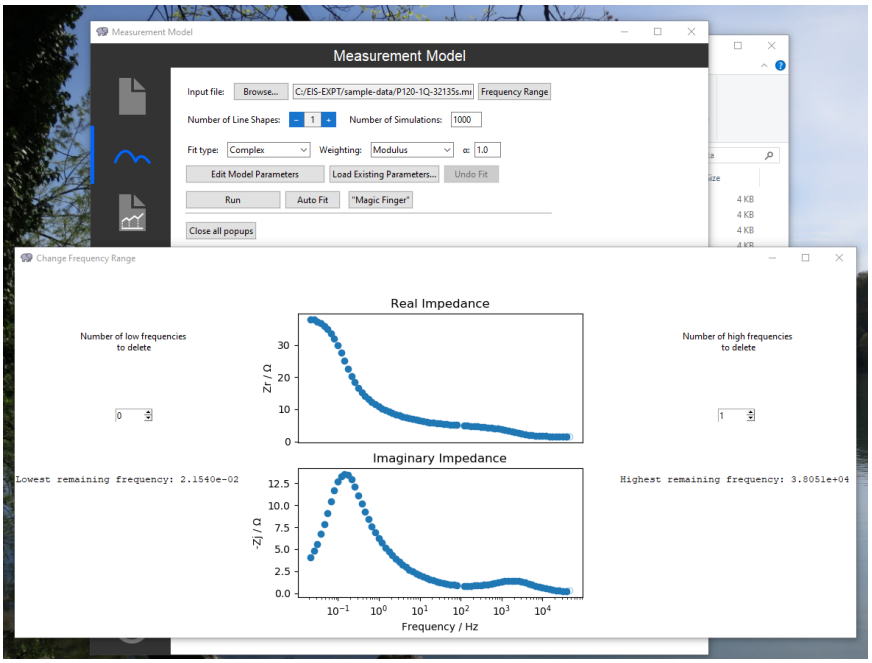

Figure 3.19: Remove the highest frequency from the sample file. The first measured frequency is often corrupted by nonstationary behavior, and the quality of the fit is improved by removing it.

We can attempt the autofit feature as shown in Figure 3.20. Nine Voigt elements are obtained, and the quality of the fit is excellent, as shown in Figure 3.21. Press the `Save Residuals for Error Analysis` button. Note that the proposed file name P120-1Q-32135s-C9.mmresiduals includes the original file name and an indication of a complex fit with nine Voigt elements.

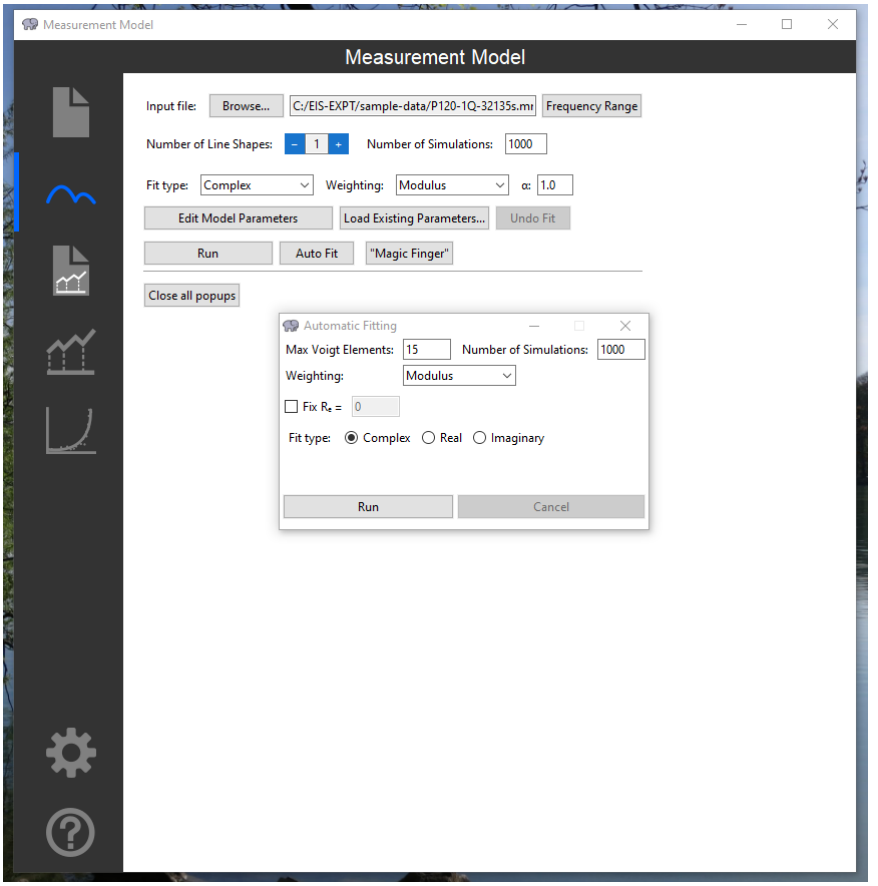

Figure 3.20: Prepare for autofit of a sample file. Modulus weighting is used in the absence of an experimental error structure.

Now create *.mmresiduals files for the other two measurements. Use the regression results from the previous regression as initial guesses for the measurement model fit to the other two files. Be careful to remove the highest frequency data point for each file before performing the regression. Please note that, while it is possible to get more Voigt elements for the P120-1Q-33305s.mmfile file than the nine obtained for P120-1Q-32135s.mmfile, we need to use the same number of Voigt elements for each *.mmresiduals file to obtain reliable estimates for the error structure. Press the `Save Residuals for Error Analysis` button after each successful regression.

To enable weighting of regressions using the experimental stochastic error structure, perform the analyses presented in Sections 4.4 and Section 5.6. The result will be parameters used as a model for the error structure.

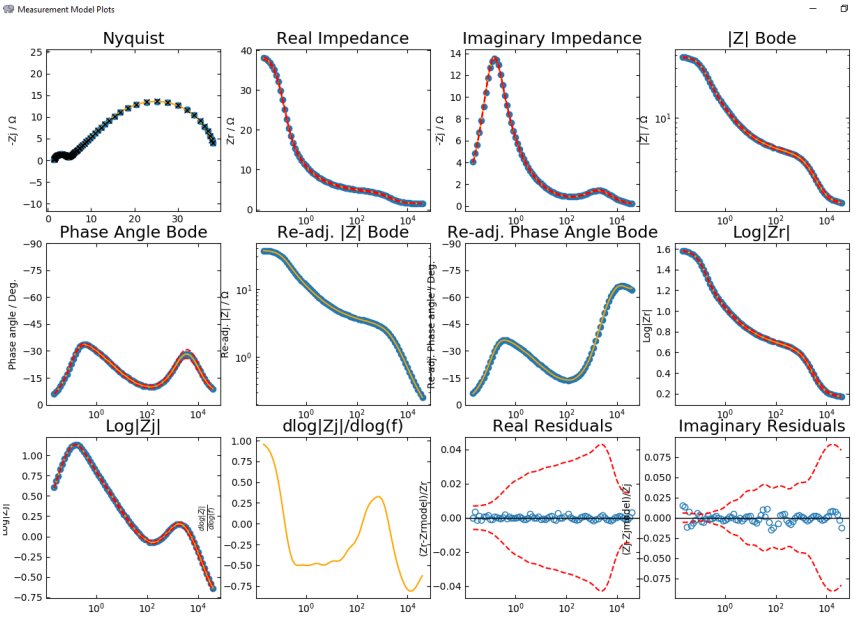

Figure 3.21: Fitting results for the regression of nine Voigt elements to the file P120-1Q-32135s.mmfile.

3.24.2 Assessment of Consistency with the Kramers–Kronig Relations

The error structure identified in Section 5.6 is used to weight regressions for subsequent analyses. The result obtained was $\alpha = 0.00088 \pm 0.00015$, $\beta = 0$, $\gamma = 3.18 \pm 0.19 \times 10^{-05}\ \Omega^{-1}\text{cm}^{-2}$, and $\delta = 0$. Load the P120-1Q-32135s.mmfile file and select the Error model weighting. Select and input the parameters α and γ. A systematic regression in which Voigt elements are added sequentially yields the output seen in Figure Use of

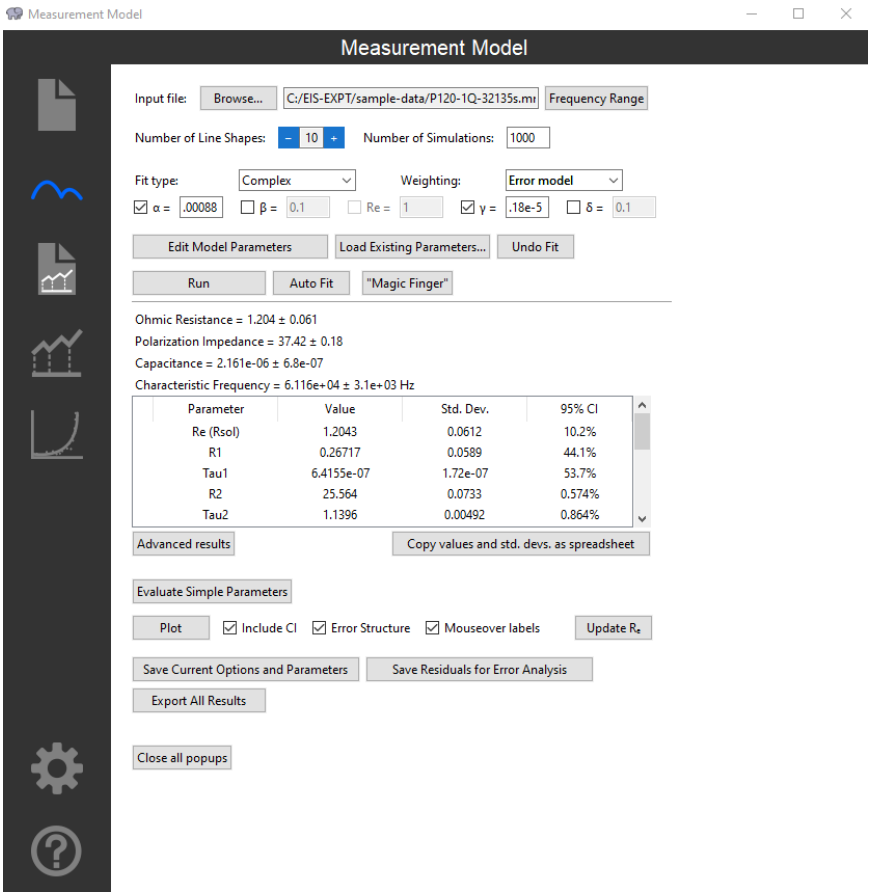

Figure 3.22: Fitting results for the regression of ten Voigt elements to the file P120-1Q-32135s.mmfile.

the experimental error structure to weight the regression yielded an additional Voigt element. The plots resulting from the regression are presented in Figure The residual errors fall within the 95.4% confidence interval for the model, suggesting that the data satisfy the Kramers–Kronig relations.

A deeper analysis can be made by fitting the measurement model to the imaginary part of the impedance. The regression allowed ten Voigt elements, and the corresponding plots are presented in Figure 3.24. The residual errors for the imaginary part of

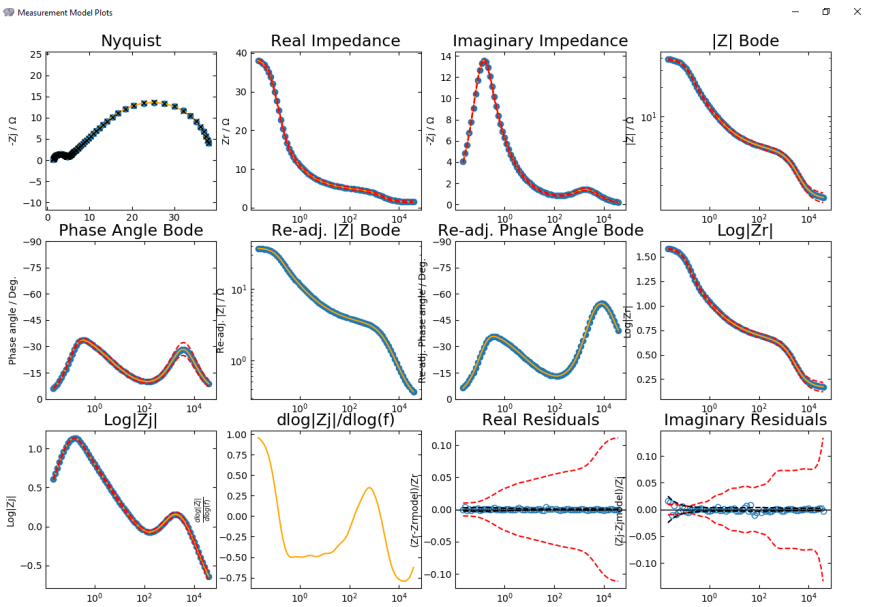

Figure 3.23: Comparison of the model to experimental data for weighted regression of a measurement model with ten Voigt elements to the file P120-1Q-32135s.mmfile. Confidence intervals for the regression are presented as red dashed lines, and the confidence interval for the data, i.e., $\pm 2\sigma$, are presented in the normalized residual error plots as black dashed lines.

the impedance fall nicely within the $\pm 2\sigma$ error bounds, i.e., the noise level of the measurement. The real residual errors appear to fall outside the confidence intervals for the model, but it must be remembered that a fit to the imaginary impedance cannot provide estimates for the ohmic resistance. Thus, the ohmic resistance is treated as an adjustable parameter.

Press the `Update Re` button. The ohmic resistance shown in Figure has a value $R_e = 1.4605$. Use the up and down arrows to adjust the ohmic resistance. An adjustment of ohmic resistance to $R_e = 1.395$ places all the real residual errors within the 95.4% confidence interval. The fit to the imaginary part of the impedance confirms that the data satisfies the Kramers–Kronig relations.

A similar analysis can be performed for the first measurement in the series. Load the file P120-1Q-14s.mmfile and remember to remove the highest frequency in the measurement. Use the error structure weighting, and add Voigt elements sequentially. The resulting panel with 10 Voigt elements is shown in Figure 3.26. A comparison of the model to experimental data is presented in Figure 3.27 for weighted regression of a measurement model with ten Voigt elements to the file P120-1Q-14s.mmfile. Confidence intervals for the data, i.e., $\pm 2\sigma$, are presented in the normalized residual error plots as black dashed lines. Clear trending is visible, suggesting that the real and imaginary parts of the impedance are inconsistent.

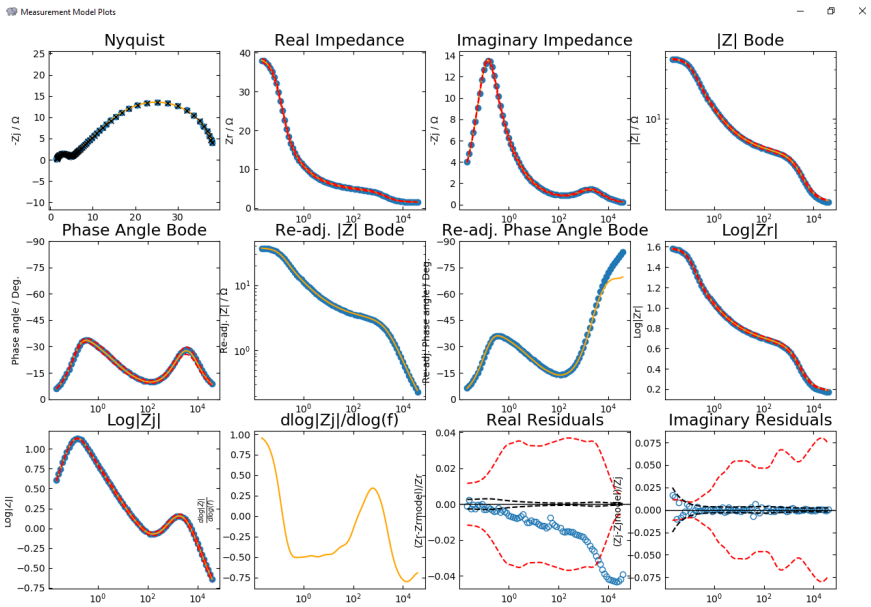

Figure 3.24: Comparison of the model to experimental data for weighted regression of a measurement model with ten Voigt elements to the file P120-1Q-32135s.mmfile. Confidence intervals for the regression are presented as red dashed lines, and the confidence interval for the data, i.e., $\pm 2\sigma$, are presented in the normalized residual error plots as black dashed lines.

A fit of a measurement model with nine Voigt elements to the imaginary part of the impedance is presented in Figure 3.28. The fit shows that, based on the argument that data that fall outside the 2σ confidence interval for the model prediction may be removed, the five lowest frequencies may be removed on the grounds that the data are not stationary. Note that more than nine Voigt elements could be fit to the data, but the standard deviations for the regressed parameters were much larger, and all data fell within the 95.4% confidence interval.

Use of the truncated frequency range will facilitate regression for the first measurement in the series. All the data satisfied the Kramers–Kronig relations for the last measurements in the series.

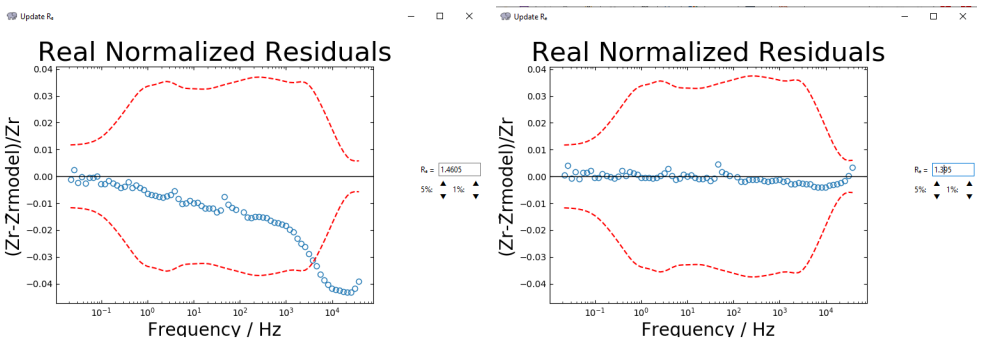

Figure 3.25: Adjustment of ohmic resistance for an imaginary fit: a) original real residual errors and b) real residual errors after adjustment of R_e.

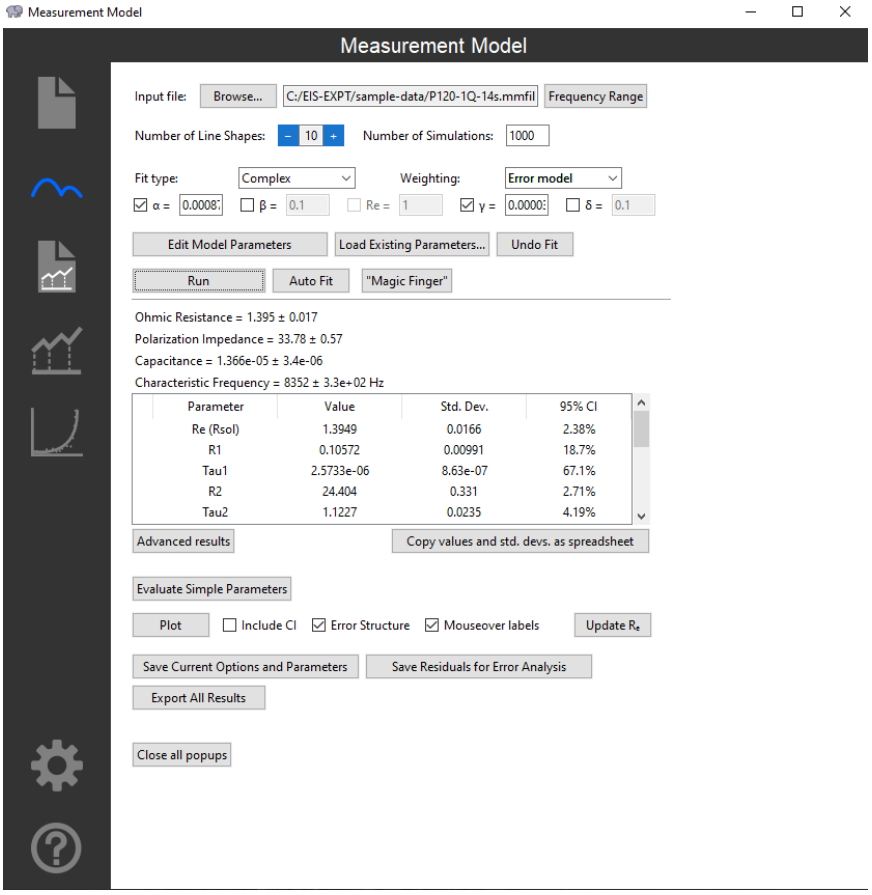

Figure 3.26: Fitting results for the regression of ten Voigt elements to the file P120-1Q-14s.mmfile.

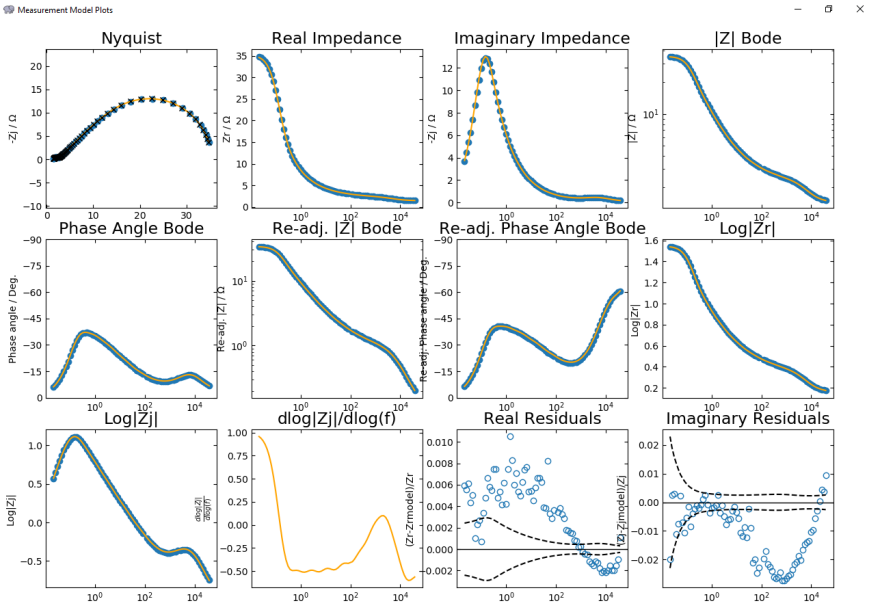

Figure 3.27: Comparison of the model to experimental data for weighted regression of a measurement model with ten Voigt elements to the file P120-1Q-14s.mmfile. Confidence intervals for the data, i.e., $\pm 2\sigma$, are presented in the normalized residual error plots as black dashed lines.

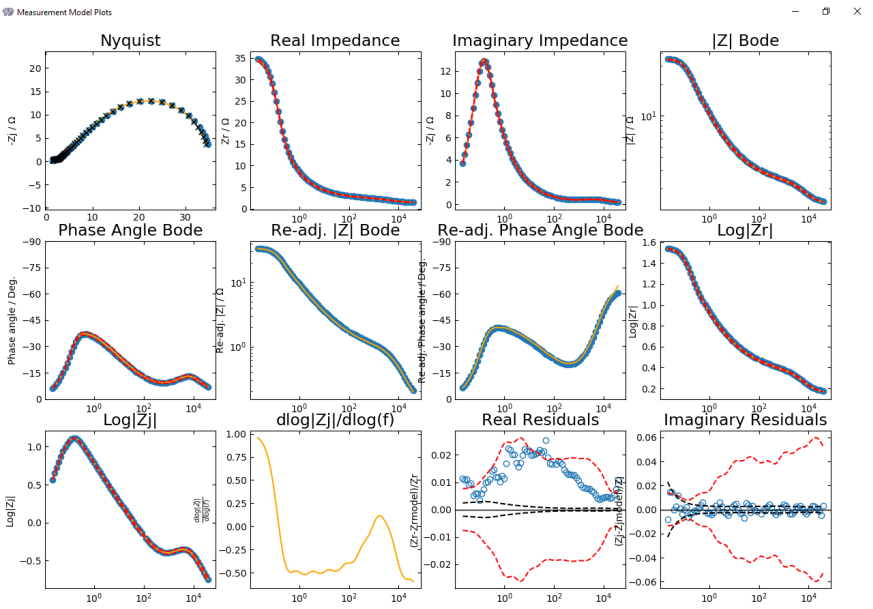

Figure 3.28: Comparison of the model to experimental data for weighted regression of a measurement model with nine Voigt elements to the file P120-1Q-14s.mmfile. Confidence intervals for the regression are presented as red dashed lines, and the confidence interval for the data, i.e., $\pm 2\sigma$, are presented in the normalized residual error plots as black dashed lines.

3.24.3 Extraction of Physical Properties

The measurement model program offers a preliminary interpretation of the impedance data by use of the `Evaluate Simple Parameters` button (see Section 3.18). Regress the measurement model to the file P120-1Q-33305s.mmfile, using error structure weighting and deleting the highest three frequencies. The image presented in Figure 3.29 shows the ohmic resistance, the polarization resistance, and the impedance at the zero-

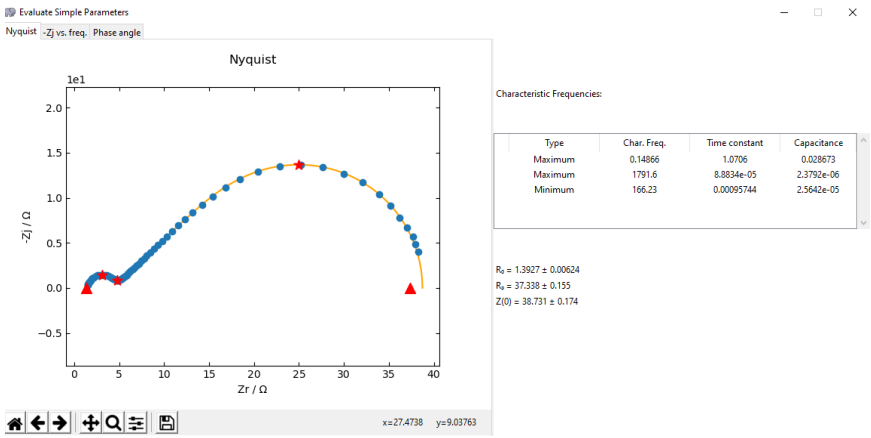

Figure 3.29: Results from the `Evaluate Simple Parameters` feature for the regression of 9 Voigt elements to the file P120-1Q-33305s.mmfile.

frequency asymptote calculated by extrapolation of the model. Figure 3.29 also shows characteristic frequencies associated with maximima and minima of $-Z_j$. The image presented in Figure 3.30 shows a slight distortion in the extrapolation to infinite fre-

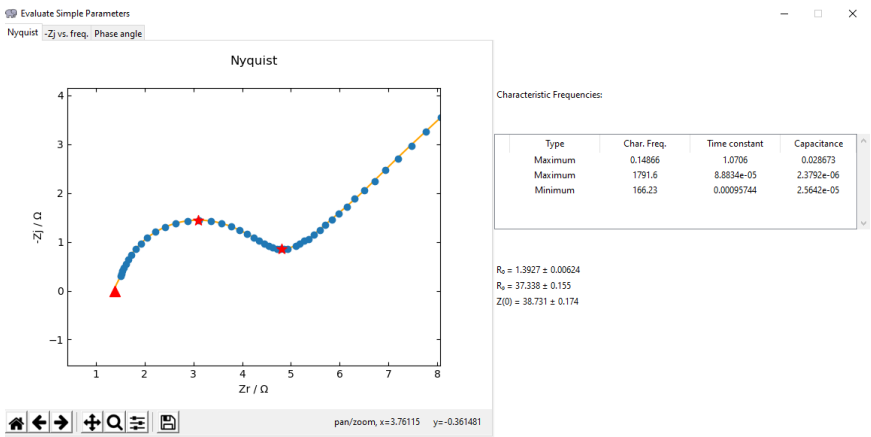

Figure 3.30: Results from the `Evaluate Simple Parameters` feature for the regression of 9 Voigt elements to the file P120-1Q-33305s.mmfile. The graphing controls were used to zoom into the high-frequency region.

Table 3.5: Parameter values obtained for the error-structure-weighted regression of 9 Voigt elements to the file P120-1Q-33305s.mmfile with a maximum frequency of 25.9 kHz.

Parameter	Value	Std. Dev.	Ωcm^2
R_e	1.3927	0.0062	Ωcm^2
$Z(0)$	38.73	0.16	Ωcm^2
Polarization Resistance	37.34	0.16	Ωcm^2
Overall Capacitance	13.16	0.70	μF/cm^2
Characteristic Frequency	8.683	0.074	kHz

quency, suggesting that the model is attempting to fit a high-frequency feature associated with the nonuniform current and potential distribution associated with the disk electrode geometry.[30,31]

The measurement model also provides values for ohmic resistance, $Z(0)$, polarization resistance, overall capacitance, and characteristic frequency calculated as

$$f_c = \frac{1}{2\pi R_e C} \tag{3.4}$$

where R_e and C are the values obtained from the measurement model parameters. It should be emphasized that the characteristic frequency should be obtained from

$$f_c = \frac{1}{2\pi R_{e,HF} C} \tag{3.5}$$

where the ohmic resistance is the high-frequency ohmic resistance $R_{e,HF}$ rather than the low-frequency ohmic resistance $R_{e,LF}$.[32,33] As the difference between $R_{e,HF}$ and $R_{e,LF}$ is typically smaller than 8 percent, equation (3.4) provides a good approximation to equation (3.5).

The results associated with Figures 3.29 and 3.30 are presented in Table 3.5 As the maximum measured frequency of 25.9 kHz is larger than the characteristic Frequency of 8.683 kHz and the electrode geometry is susceptible to nonuniform current and potential distributions, the regression shown in Table 3.5 includes the influence of an ohmic impedance.

Sequentially remove high frequency data until the maximum measured frequency is lower than the characteristic frequency identified by the program. Note that the characteristic frequency will change as high-frequency data are truncated as the ohmic resistance and capacitance will change as the influence of the ohmic impedance is removed. After the highest 13 frequencies are removed, i.e., with a maximum frequency of 3.8 kHz, the resulting parameters are those presented in Table 3.6. The ohmic resistance given in Table 3.6 may be considered to be an estimate for $R_{e,LF}$, and the ohmic resistance given in Table 3.5 may be considered to be an estimate for $R_{e,HF}$. For the present case, $R_{e,LF}/R_{e,HF} = 1.084$, and the capacitance of $C = 24.7 \pm 2.9$ μF/cm^2 is in the range typical of a double layer capacitance.

Table 3.6: Parameter values obtained for the error-structure-weighted regression of 9 Voigt elements to the file P120-1Q-33305s.mmfile with a maximum frequency of 3.8 kHz.

Parameter	Value	Std. Dev.	Ωcm^2
R_e	1.5098	0.0075	Ωcm^2
$Z(0)$	38.76	0.52	Ωcm^2
Polarization Resistance	37.25	0.52	Ωcm^2
Overall Capacitance	24.7	2.9	$\mu\text{F/cm}^2$
Characteristic Frequency	4.276	0.079	kHz

Chapter 4

Error File Preparation

Prior to fitting an error structure under the Error Analysis tab (see Chapter 5), the **.mmresiduals* files saved from fittings to repeated impedance measurements must be combined into a **.mmerrors* file. This can be accomplished in the Error File Preparation tab. To access this tab, click the file icon with an error graph in the middle ( ). Figure 4.1 shows the tab before any files have been loaded.

> **Warning!** It is important that all files combined into a **.mmerrors* file have been fit with the same number of statistically significant parameters. Likewise, all files must have the same frequencies.

> **Tip!** If the same number of parameters could not be obtained for each data file, either reduce the number of parameters to the lowest common value or eliminate the file yielding a smaller number of parameters.

4.1 File Input

To begin, use the `Add File(s)` button to choose one or more **.mmresiduals* files. Once the file dialog is confirmed, the file name(s) will appear in the listbox to the upper right of the tab. If more than one file is loaded, the listbox below will display the average values and standard deviations of the residuals at each frequency, and the `Plot` and `Save errors` buttons will become active (see Figure 4.2).

The `Real Mean` is the average of the real residual errors at a given frequency, and the `Real Std. Dev.` is the standard deviation of the real residuals between the loaded files at that frequency. Once one file is loaded, a line will appear under the listbox displaying the average ohmic resistance and average overall capacitance of the loaded files.

If files do not have the same number of frequencies, or if the files do not have the same frequency values, an error will occur. If a file is loaded that has already been loaded, a warning will appear but the file will still be loaded.

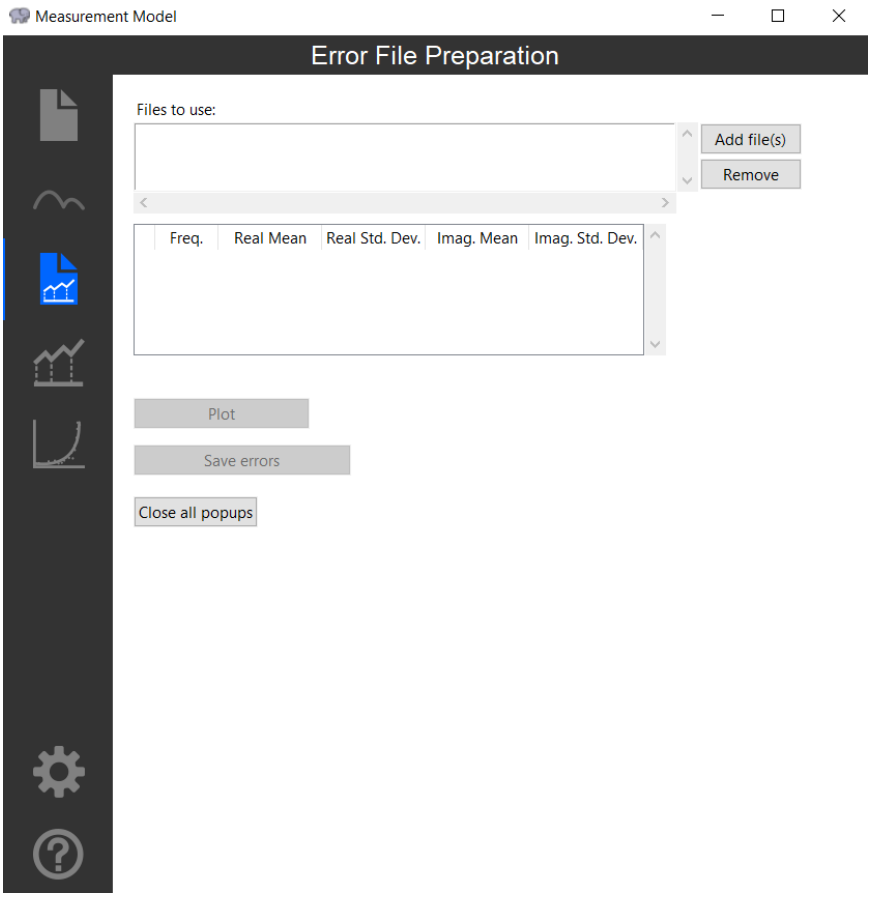

Figure 4.1: The Error File Preparation tab. More actions become available once a series of **.mm-residuals* files are loaded.

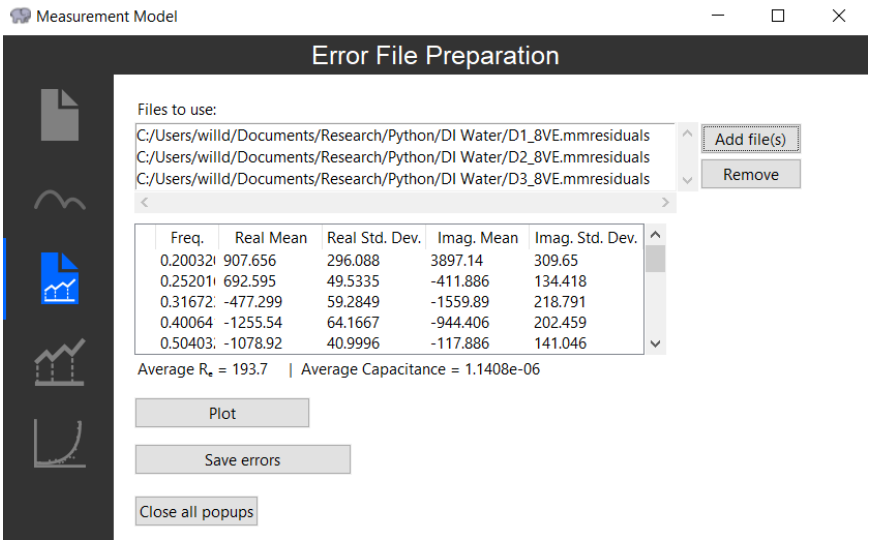

Figure 4.2: The Error File Preparation tab after loading three residuals files for a Pt electrode in deionized water, each fit with eight Voigt elements. The listbox includes values for frequency, mean value and standard deviation for the real residual error, and mean value and standard deviation for the imaginary residual error. The `Plot` and `Save errors` buttons are activated.

In order to remove a file from the list, click the desired file so it is highlighted in the listbox, then click the `Remove` button or right-click and click `Remove file`.

4.2 Plot

Once at least two files are added, the `Plot` button will become active. This button will bring up a plot of the overlayed real and imaginary standard deviations of the residuals (i.e., columns 3 and 5 of the listbox shown in Figure 4.2) as functions of frequency, as shown in Figure 4.3. The real standard deviations are represented as blue circles, and the imaginary are represented as orange triangles.

4.3 Save Errors

Once the desired **.mmresiduals* files have been loaded, a new **.mmerrors* file can be saved by using the `Save errors` button. This file will include the frequencies, real and imaginary standard deviations of the residuals, the real and imaginary impedance data, and the standard deviations between the real and imaginary standard deviations at a given frequency (i.e., σ_σ).

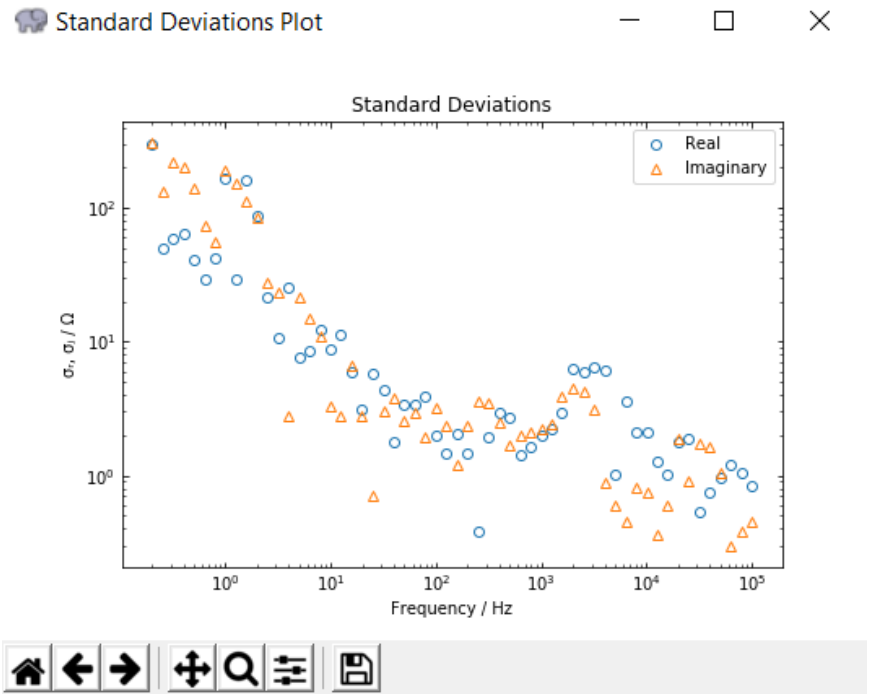

Figure 4.3: Standard deviations of residuals for three measurements for a Pt electrode in deionized water fit with eight Voigt elements. Inequality of the real and imaginary parts of the impedance at any given frequency range may suggest that the Kramers–Kronig relations are not satisfied.

4.4 Application to Sample Data

Open the Error File Preparation tab, and add the *.mmresiduals files prepared in Section 3.24. If the data sets do not include the same number of frequencies, an error message "Error 27: The number of data do not match" will be reported. The system does not catch inconsistencies in the numbers of Voigt elements used, so the user needs to be careful. The resulting standard deviations are shown in Figure 4.4. The standard deviations for the real part of the impedance (◯) and the standard deviations for the imaginary part of the impedance (△) overlap, suggesting that the data are consistent with the Kramers–Kronig relations. Save the standard deviation file by pressing the `Save errors` button.

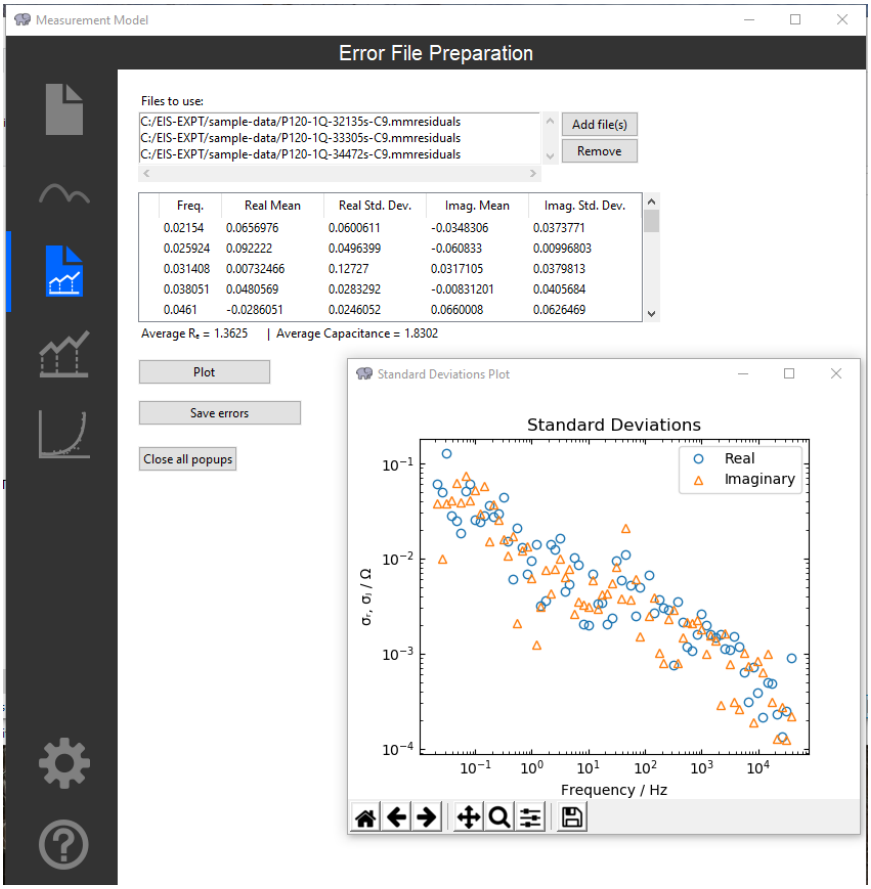

Figure 4.4: The error file preparation tab and results for the sample data. Standard deviations for the real (◯) and the imaginary (△) parts of the impedance overlap, suggesting that the data are consistent with the Kramers–Kronig relations.

Chapter 5

Error Analysis

The principle behind using the measurement model to estimate the stochastic contribution to the error structure was presented by Agarwal et al.[2] The approach has been applied to a number of electrochemical[34–38] and non-electrochemical systems.[11,39] The error analysis tab can be accessed with the error graph icon ( ) in the navigation pane as shown in Figure 5.1.

5.1 File Input

The file input is very similar to that used in the Error File Preparation tab described in Section 4.1. Use the `Add File(s)` button to choose one or more **.mmerrors* files. Once the file dialog is confirmed, the file name(s) will appear in the listbox to the upper right of the tab.

In order to remove a file from the list, click the desired file such that it is highlighted in the listbox, then click the `Remove` button or right-click and click `Remove file`.

5.2 Error Structure

The complete error structure model is of the form

$$\sigma = \alpha \left|Z_j\right| + \beta \left|Z_r - R_e\right| + \gamma \left|Z\right|^2 + \delta \tag{5.1}$$

The `Error structure model` line beneath the file listbox shows the current error structure to be fit. This can be modified by activating or deactivating the checkboxes below: checking a box will cause that parameter to be added to the fitting and will update the `Error structure model` line. If β is checked, the R_e checkbox will appear below it; if unchecked, R_e will disappear. The default error structure choice is $\gamma \left|Z\right|^2 + \delta$.

> **Tip!** The model which includes the role of R_e in equation (5.1) appears to give a better representation of the standard deviation for systems which

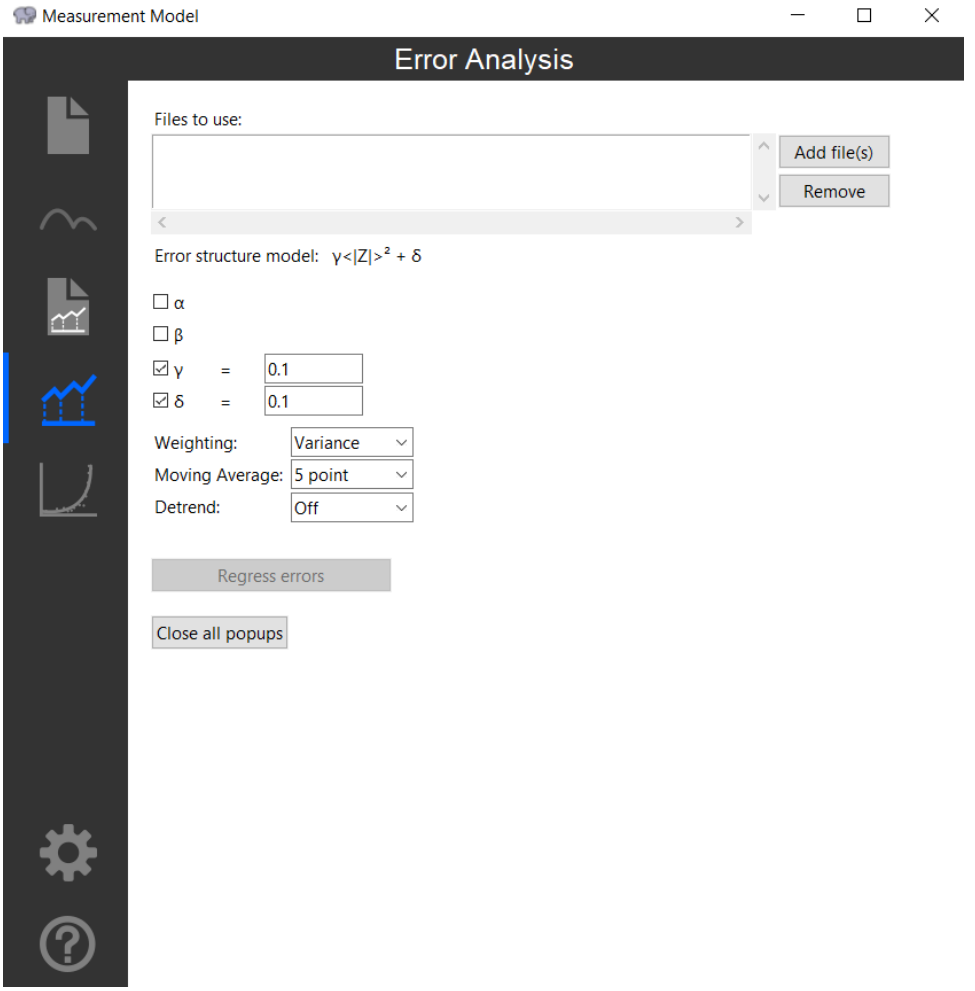

Figure 5.1: The Error Analysis tab. The `Regress errors` button will be available after **.mmerrors* files are added.

have a large solution resistance, but is more awkward to use because it requires an a priori estimate of R_e. For systems that have a relatively small solution resistance, it is sufficient to ignore this contribution.

Equation (5.1) is an empirical relation that was developed initially for the Solartron 1286 potentiostat coupled with Solartron 1250 frequency response analyzer.[39] It has been used successfully for a variety of impedance instrumentation. Typically, the impedance is a strong function of frequency and can vary over several orders of magnitude in the experimentally accessible frequency range. The stochastic errors of the impedance measurement are strongly heteroscedastic, which means, in this case, that the standard deviation of the stochastic part of the measurement depends on frequency.[6] By making the model for standard deviation be a function of the impedance, we account for the dependence on frequency.

5.3 Fitting Options

Due to the heteroscedastic nature of the standard deviation, it is important to choose the proper options when fitting in order to obtain a good regression.

5.3.1 Weighting

The weighting used for the regression can have a significant influence because the standard deviation is a very strong function of frequency. The `Weighting` dropdown provides options: `None` means that no weighting will be used (all weights will be unity) and `Variance` (the default) means that each frequency will be weighted by the standard deviation between the real and imaginary parts of the standard deviations of the residuals. If `None` is chosen, the `Moving Average` dropdown will disappear.

5.3.2 Moving Average

The weighting applied under the variance option is calculated under the assumption the real and imaginary standard deviations are two samples of the same value. The variances used in weighting can fluctuate between nearby frequencies. In order to smooth the weighting, a moving average can be taken using nearby points. This can be done using the `Moving Average` dropdown. `None` means that no moving average will be taken. `3 point` means that the variance will calculated using real and imaginary standard deviations at frequencies immediately before and after the present frequency. `5 point` (the default) means that the variance will be averaged using real and imaginary standard deviations at two frequencies before and two frequencies after the present frequency. The `Moving Average` dropdown is only available if `Variance` weighting is chosen.

5.3.3 Detrend

Fitting can potentially be improved by centering all data around zero. Detrending can accomplish this by subtracting the average value of a series from every point in that series. This can be done using the `Detrend` dropdown. `Off` (the default) means that no detrending will be used; `On` means that all standard deviations and impedance values will be detrended.

> **Tip!** Try fitting with and without detrending to see which yields a better fit and better confidence intervals on the parameters.

5.4 Regress Errors

The `Regress errors` button will use a Levenberg–Marquardt regression to fit the chosen parameters using the current settings. If a fitting fails, an alert will appear. Otherwise, the results will appear below.

Both the real and imaginary standard deviations are fit at the same time. The equality of the real and imaginary parts of the impedance is now supported by a large amount of experimental evidence,[39,36,40] by theoretical arguments based on the Kramers–Kronig relations,[11,41,42] and by arguments based on propagation of errors from time to frequency domain.[43,44] Shukla et al.[14] showed that the error structure obtained was independent of the mathematical form of the error structure used.

5.5 Results

After a regression has completed successfully, the resulting parameter values will be shown in a listbox beneath the `Regress errors` button. Figure 5.2 shows a successful error structure fit. Parameters with confidence intervals greater than 100% include zero as a possible value and are not considered statistically significant. Such parameters are highlighted yellow, as shown in see Figure 5.3. Parameters that are not statistically significant should be removed from the fitting. While the regression program permits parameters to have negative values and does not highlight these results, such parameters should be removed on the grounds that there should be no negative contributions to a standard deviation.

> **Tip!** Continue to delete or add model parameters α, β, γ, and δ until the regression yields positive parameters with 95.4% confidence intervals smaller than 100% of the parameter value.

The procedure for selecting an error structure model is a trial-and-error process. Often, a common error-structure model can be found for measurements performed using a given impedance system and measurement strategy. For this reason, there is an advantage to seeking an error structure model for a group of **.mmerrors* files.

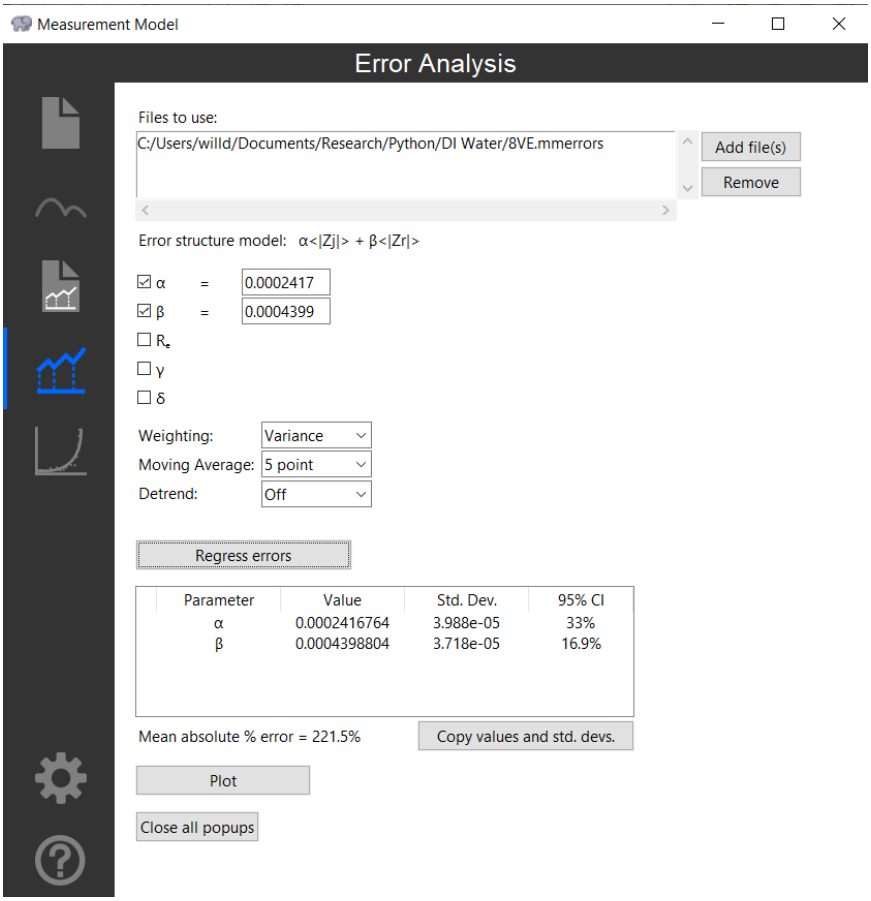

Figure 5.2: A successful fit of the error structure $\alpha\left|Z_{\mathrm{j}}\right|+\beta\left|Z_{\mathrm{r}}\right|$ for a Pt electrode in deionized water originally fit with eight Voigt elements.

Parameter	Value	Std. Dev.	95% CI
α	0.0002244399	4.221e-05	37.6%
β	0.000553278	0.0001023	37%
δ	-0.3796469	0.3194	168%

Figure 5.3: An unsuccessful fit of the error structure $\alpha\left|Z_{\mathrm{j}}\right|+\beta\left|Z_{\mathrm{r}}\right|+\delta$. The error parameter δ should be rejected for two reasons: it has a negative value and the 95.4% confidence interval includes zero.

5.5.1 Mean Absolute Percent Error

As the weighting and detrending choices can have a significant impact on the χ^2 value, the `Mean absolute % error` is reported instead. This value is found by taking the average of the absolute values of the percent differences between the model and the real and imaginary standard deviations at each frequency. This value should be comparable between different fittings; a lower value indicates a "better" fit.

5.5.2 Copy Values and Std. Devs.

The `Copy values and std. devs.` button will copy the files names, fitting settings, parameter values and standard deviations, and correlation matrix in a format that can be easily pasted into a spreadsheet program.

5.5.3 Plot

The `Plot` button will bring up a plot for each file fit of standard deviations against frequency, shown in Figure 5.4. The plot will include the real and imaginary standard devations as blue circles and orange triangles, respectively. The error structure model will be plotted as a green line.

> **Tip!** In many, but not all cases, a common error structure can be identified which seems to be a function only of the instruments used. It is worthwhile to attempt to find such a common set of parameters because the resulting model can give a good estimate for the error structure of other data collected under the same conditions.

> **Tip!** A large error structure may signify a need to improve experimental parameters or conditions.

> **Tip!** For measurements that do not introduce a bias error, the standard deviation of the real part of the measurement should be equal to the standard deviation of the imaginary part. Deviation from this result may arise from problems with the experiment.

5.6 Application to Sample Data

A model for the error structure can be obtained in the Error Analysis tab. Open the file generated in Section 4.4. The selection of the best model is a trial-and-error procedure. Use variance weighting with a five-point moving average. The result obtained with the detrend option enabled was $\alpha = 0.00088 \pm 0.00015$ and $\gamma = 3.18 \pm 0.19 \times 10^{-05}\ \Omega^{-1}\text{cm}^{-2}$, where

$$\sigma = \alpha|Z_j| + \gamma|Z|^2 \tag{5.2}$$

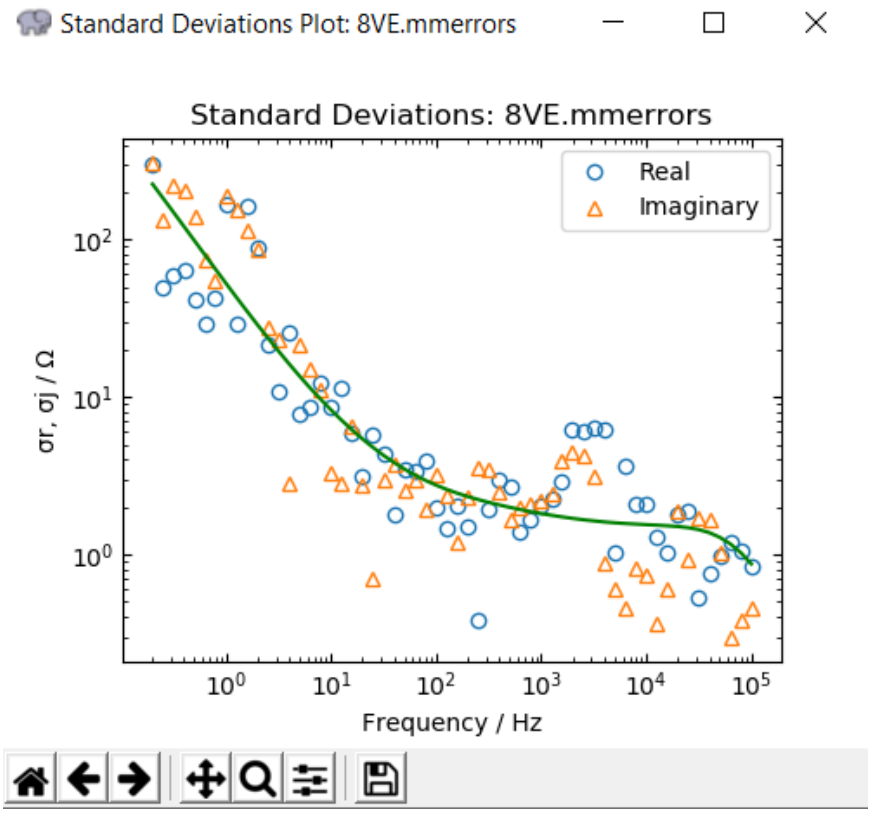

Figure 5.4: The error structure plot for a fitting of $\alpha\left|Z_{\mathrm{j}}\right| + \beta\left|Z_{\mathrm{r}}\right|$ to data for a Pt electrode in deionized water. The line captures the frequency dependence of the real and imaginary standard deviations.

The units of γ depend on the units of the impedance being analyzed. The resulting tab view is shown in Figure 5.5, and the comparison of the model and the standard deviations is shown in Figure 5.6. The `Copy values and std. devs.` button will copy the files names, fitting settings, parameter values and standard deviations, and correlation matrix in a format that can be easily pasted into a spreadsheet program.

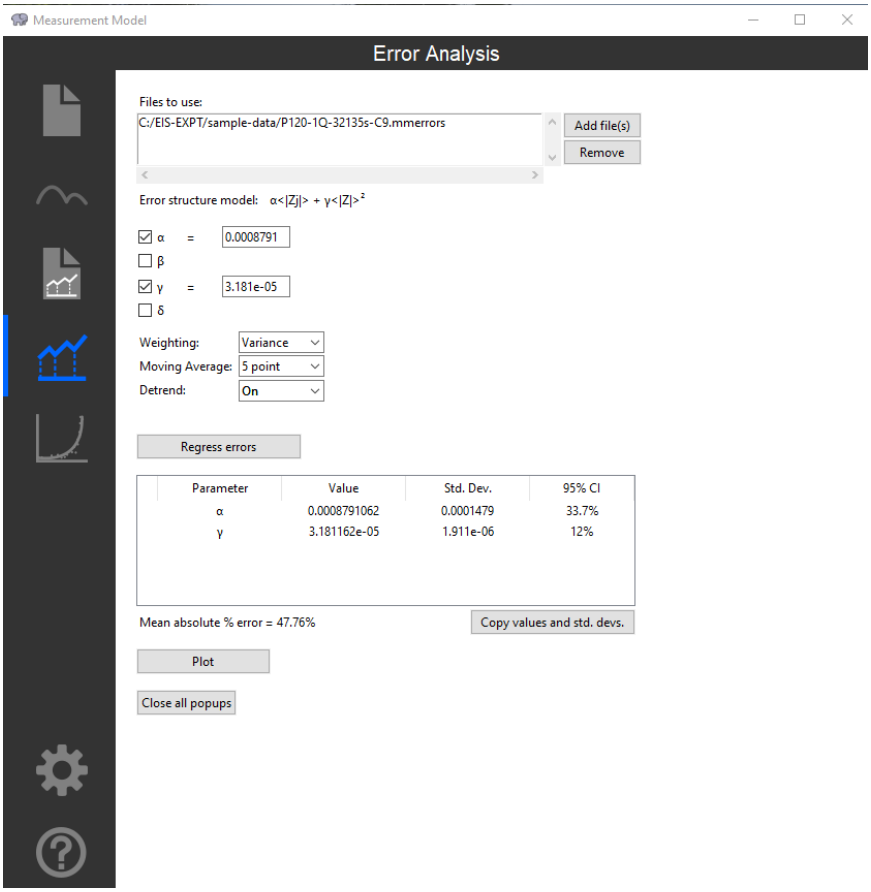

Figure 5.5: The error model tab and results for the sample data set. The model has the form shown in equation (5.2) with parameters $\alpha = 0.00088 \pm 0.00015$ and $\gamma = 3.181162 \pm 0.19 \times 10^{-05}\ \Omega^{-1}\text{cm}^{-2}$.

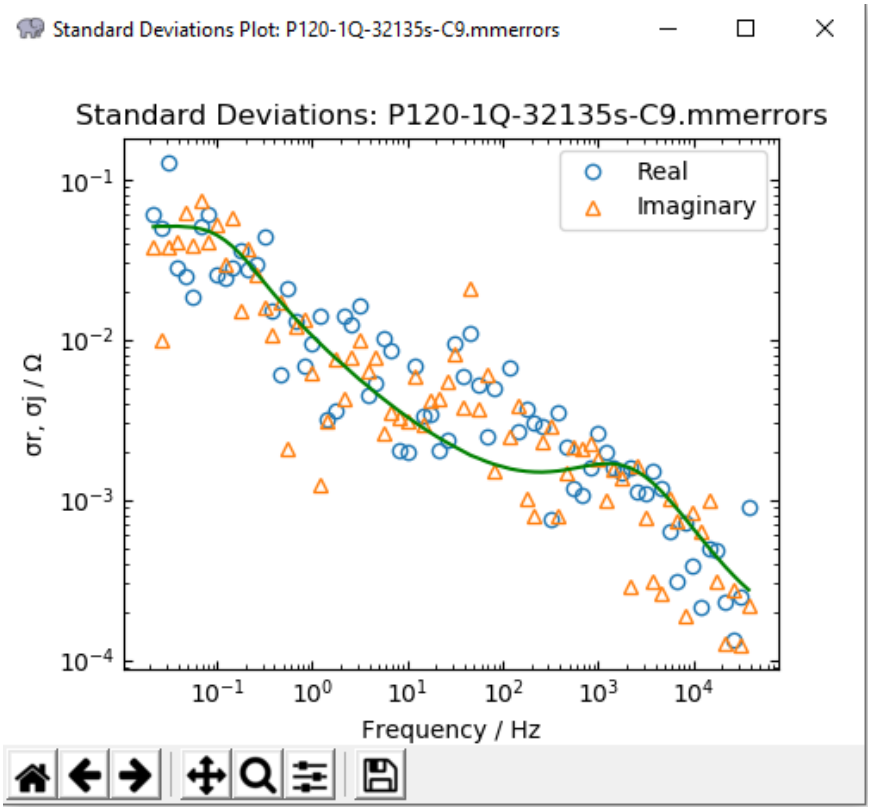

Figure 5.6: Plot showing the results presented in Figure 5.5.

Chapter 6

Custom Formula Fitting

The Measurement Model program also has the capability to fit user-defined models. To avoid limitations associated with models defined in terms of electrical circuit analogs, this program is equation-based. A user-defined formula can be fit in the Custom Formula Fitting tab shown in Figure 6.1, accessible with the graph icon ( ). The program requires the formula to be input as Python code. There are many free, high-quality resources online to learn the basics of Python syntax.

> **Warning!** Almost all Python code can be typed in and executed. Please note that this code could potentially damage the main program or even the computer!

6.1 File Input

A file can be loaded using the `Browse...` button at the top left of the tab. Available file types include **.mmfile* and **.mmcustom*. After loading a file successfully, its name will appear in the textbox to the right. If a **.mmcustom* file is chosen, the program will load the corresponding linked **.mmfile* as well as code, fitting choices, and fitting parameters.

6.2 Fitting Options

The `Frequency Range` button, `Fit type` dropdown, `Number of Simulations` textbox, and α textbox act exactly as they do in the Measurement Model tab described in Sections 3.3, 3.6, 3.5, and 3.7.1, respectively.

The `Weighting` dropdown acts almost exactly as it does in Section 3.7. However, it includes an additional option of `Custom`. If this is chosen, the weighting used will be taken from a variable called `weighting` in the code. This weighting will be the same for the real and imaginary parts at a given frequency.

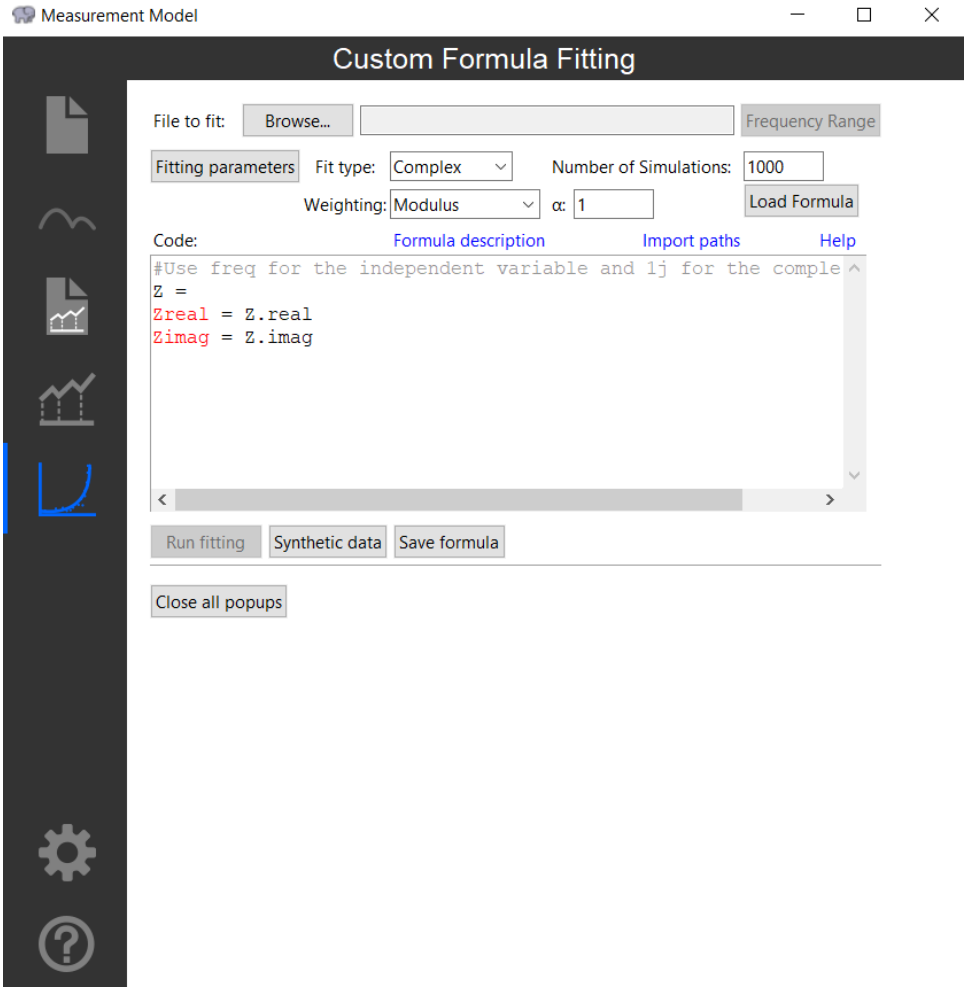

Figure 6.1: The Custom Formula Fitting tab. The Code textbox is preloaded with the structure required to interface with the regression program.

6.3 Custom Formula Definition

Custom Python code can be defined in the `Code` box. The program makes available three pre-defined lists: `Zr`, `Zj`, and `freq`, which are the real impedance values, the imaginary impedance values, and the frequencies, respectively. The program expects to find two lists: `Zreal` and `Zimag`, in which are held the real and imaginary model impedances, respectively. If `Custom` weighting has been chosen, then the program will expect a third variable: `weighting`.

Python can operate on whole lists/arrays at once, meaning loops are often unnecessary and code can be short. Note that the symbol \can be used to break long lines of code and continue on the next line.

6.3.1 Fitting Parameters

All fitting parameters must be added using the `Fitting parameters` button, which will open a popup to display all current parameters, as shown in Figure 6.2.

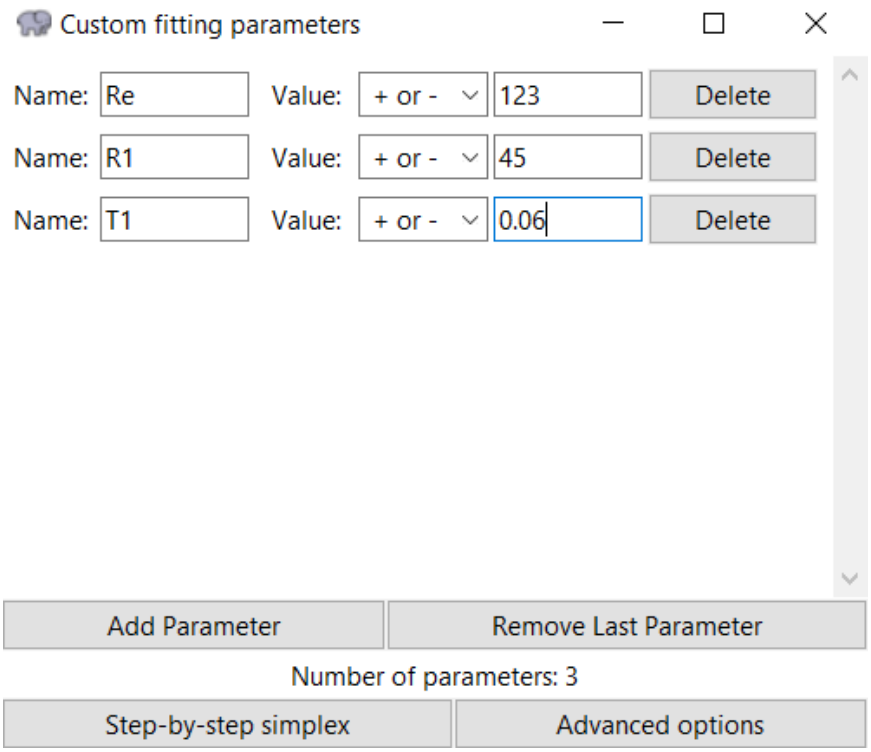

Figure 6.2: Three named custom parameters with initial guesses. As parameter names are defined, the parameter color will change in the Code textbox. Parameter names may not be Python-reserved words nor variable names used in the Measurement Model.

At first, the popup will be empty. Add a parameter with the `Add Parameter` button. This will add a new line to the popup with a new parameter. The parameter name defined in the leftmost textbox should be the same as is used in the code. The dropdown to the right can be used to restrict the parameter value, and the textbox to the right holds the initial guess for the parameter. The `Delete` button can be used to remove a given parameter, and the `Remove Last Parameter` button will remove the parameter farthest down the popup. These parameters cannot be recovered once removed. The `Step-by-step simplex` button will become active once a file is loaded (see Section 6.3.2), and the `Advanced options` button will open a popup containing further choices for each parameter, as described in Section 6.3.3.

Parameter names must be alphanumeric, cannot be duplicates, cannot be any of the following predefined names: `freq`, `Zr`, `Zj`, `Zreal`, `Zimag`, or `weighting`, and cannot be

a Python-reserved word, i.e.,

False	class	finally	is	return
None	continue	for	lambda	try
True	def	from	nonlocal	
and	del	global	not	while
as	elif	if	or	with
assert	else	import	pass	
break	except	in	raise	yield

6.3.2 Step-by-Step Simplex

The `Step-by-step simplex` button will perform the first "step" in the Nelder-Mead simplex algorithm using the current code and loaded file. The initial simplex is constructed using the current parameter values, and the fitting will respect fixed parameters and parameter constraints. This feature can be used as a more robust method for determining appropriate initial guesses for use with the main Levenberg–Marquardt algorithm.

6.3.3 Advanced Options

The `Advanced options` button will open a popup containing further choices for each parameter, as seen in Figure 6.3. At the top are lower and upper limits; the fitting will not allow the parameter's value to go below the lower limit, or above the upper limit. The values must be real numbers (with the upper limit being greater than the lower); however, `-inf` and `inf` are accepted as well. The terms `-inf` and `inf` mean no lower or upper limit, respectively. The dropdown by the parameter values in the main parameter popup can also control the upper and lower limits. `Fixed` will disable both the upper and lower limit box, `-` will make the upper limit `0` and the lower limit `-inf`, `+ or -` will make the upper limit `inf` and the lower limit `-inf`, `+` will make the upper limit `inf` and the lower limit `0`, and `Custom` will open the advanced options popup. Changing the values in the advanced options popup itself will set the dropdown to `Custom`, unless the values are a combination listed above (for instance, setting the upper limit to `inf` and the lower limit to `-inf` will set the dropdown's value to `+ or -`).

The multistart options work exactly as they do in the Measurement Model tab (see Section 3.9). The options displayed here apply to individual parameters. A global selection of limits and multistart parameters is not allowed.

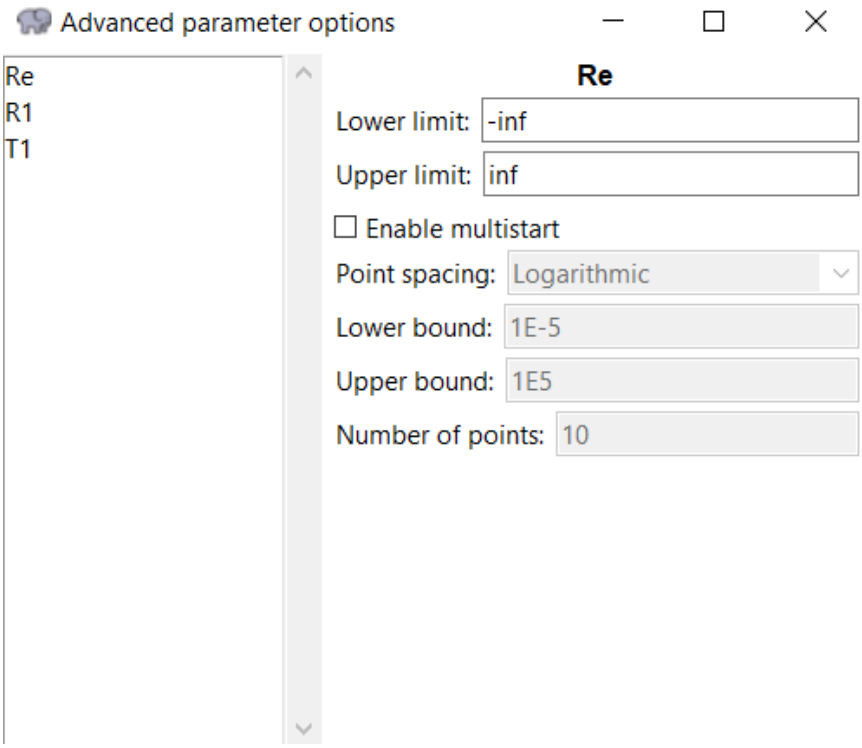

Figure 6.3: Advanced options popup for custom fitting. The options displayed here apply to individual parameters: global selection of limits and multistart parameters is not allowed.

6.3.4 Built-Ins

While Python packages can be imported for more complex operations, `numpy` is already imported as `np`. As in standard Python, the imaginary number can be accessed as `1j`. A number of functions are already included. These can be seen in Table 6.1. Practically

Table 6.1: Included functions. Note that these functions are case-sensitive. In other words, use of the Matlab® or Origin® command "pi", as will not yield the same result as `PI`.

Function (used in code)	Meaning
`PI`	$\pi = 3.141592653589793$
`SQRT`	$\sqrt{x}$
`ABS`	$\lvert x \rvert$
`EXP`	e^x
`SIN`	$\sin(x)$
`COS`	$\cos(x)$
`TAN`	$\tan(x)$
`ARCSIN`	$\sin^{-1}(x)$
`ARCCOS`	$\cos^{-1}(x)$
`ARCTAN`	$\tan^{-1}(x)$
`SINH`	$\sinh(x)$
`COSH`	$\cosh(x)$
`TANH`	$\tanh(x)$
`ARCSINH`	$\sinh^{-1}(x)$
`ARCCOSH`	$\cosh^{-1}(x)$
`ARCTANH`	$\tanh^{-1}(x)$
`LN`	$\ln(x)$
`LOG`	$\log_{10}(x)$
`RAD2DEG`	Converts radians to degrees: $x * 180/\pi$
`DEG2RAD`	Converts degrees to radians: $x * \pi/180$

Note: (Arc)trig functions take/return arguments in radians

any other mathematical operation can be using Python libraries, references for which can be found online; see, for example, integration and differential equations.

6.3.5 Import Paths

As mentioned above, common python modules can be imported by simply using the `import` keyword. However, the Python interpreter will need to know where to locate custom-written or otherwise less-common modules. This can be done either by placing the Python file in the same directory as the actual executable (not its shortcut or its installer) or by using the `Import paths` link above the code textbox. Clicking this will open the popup shown in Figure 6.4.

To the left is a listbox where the current extra import paths will be listed. To the right are two buttons. The `Browse...` button will open a dialog where the directory

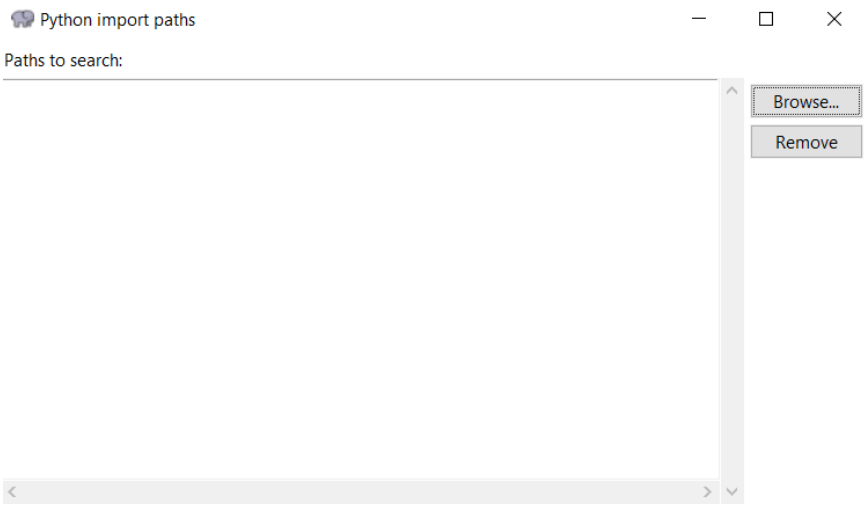

Figure 6.4: The extra import paths popup. The equation-based custom formula fitting allows use of more sophisticated code that can be imported. It can be convenient to put these supporting codes in a separate directory. This popup allows selection of the supporting code location.

to be searched can be chosen; the once this directory is selected, it will appear in the listbox to the right, and the interpreter will look for modules in it. After a directory is listed, it can be removed by either selecting it and clicking the `Remove` button, or by right clicking and pressing `Remove directory`.

For example, say one wished to create a fitting utilizing a look-up table program written as a separate Python module, and that the look-up table program file was called `LUT.py` and had 194 lines of code. To simplify the presentation in the Code textbox, the module could be used by simply calling `import LUT` in the main code. The directory where `LUT.py` is located should be loaded into the `Import paths`.

6.3.6 Syntax Highlighting

Code is automatically syntax highlighted. Supplied variables (`Zr`, `Zj`, and `freq`), fitting variables, and result variables (`Zreal`, `Zimag`, and `weighting`) are highlighted red. Included functions, the imaginary number, and Python built-in functions are highlighted purple. Strings are highlighted green, comments are highlighted gray, and Python reserved words are highlighted blue.

An example of a custom fitting formula using the parameters seen in Figure 6.2 is shown below:

```
Z = Re + R1/(1+1j*2*PI*freq*T1)
Zreal = Z.real
Zimag = Z.imag
```

The formula corresponds to an ohmic resistance in series with parallel connection of a resistor and a capacitor. The variables are `Re`, `R1`, and `T1`, where `T1=R1*C1`.

6.3.7 Save Fitting

The `Save fitting` button will save the current fitting options, fitting parameters, code, description, equation, and a link to the loaded **.mmfile* as a new **.mmcustom* file.

6.4 Formula Files

The `Save formula` button can be used to save the current formula; the information saved includes the parameter names, values, choices (such as limits and multistart options), as well as the current code, extra import paths, description, and equation. This information is saved in a file with the extension **.mmformula*. The default save name is that of the current loaded formula, if any, and the default save directory is the formula directory. Note that **.mmformula* files are identical to **.mmcustom* files in every respect; the different extensions are merely used to indicate their different uses.

6.4.1 Formula Description

The `Formula description` link above the code box can be clicked to open a popup. The textbox at the top of this popup can be used to hold a description; it can be helpful to talk about the code, parameters, or theory. Below the description textbox is an equation textbox. This accepts LaTeX-style math formatting to produce an equation, which is displayed at the bottom of the window. The textbox ignores line breaks, but \\ can be used to create a line break in the displayed equation. Consult online guides for help with LaTeXmath formatting. An example equation is shown in Figure 6.5.

6.4.2 Load Formula

The `Load Formula` button can be used to import an existing **.mmformula* file through a popup window shown in Figure 6.6. The default top-level path is as specified in Settings (see Chapter 7). This path can be changed by clicking the `Browse...` button. The top-level path will appear at the top of the listbox to the window's left. The directory can be expanded by double-clicking it or by clicking the `+` to the left. This will display all subdirectories (which can be further expanded), alongside all **.mmformula* files. The names are arranged in alphabetical order. Clicking a **.mmformula* file will load its description, code, and equation in the boxes to the right.

Once a formula has been selected, it can be loaded by clicking the `Load Formula` button at the bottom of the popup. The checkboxes to the right control what is loaded: `Parameters` will load the parameter names, values, and choices; `Code` will load the code; and `Other fitting settings` will load the number of Monte Carlo simulations, the weighting, and the fit type. `Parameters` and `Code` are selected by default.

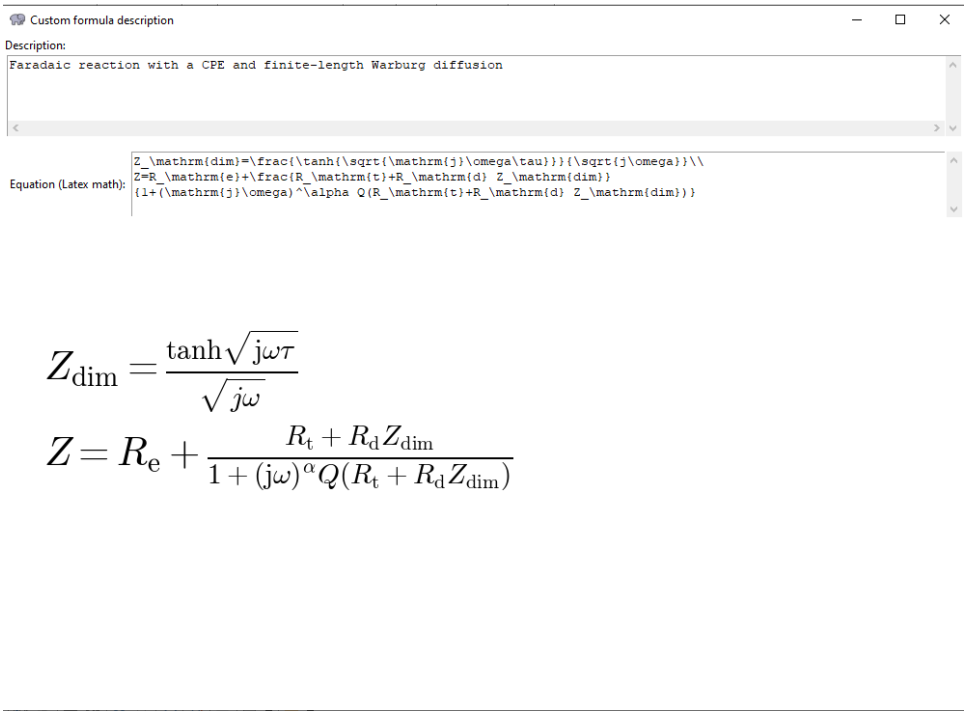

Figure 6.5: The Custom Formula Description popup showing an editable text box that can be used to write a description of the model and a box that can be used to write the equation corresponding to the model using LaTeXcommands. Note the use of \\ to separate lines of equations.

> **Warning!** Loading a formula will overwrite any existing code and parameters without prompting, and these will not be retrievable unless they were saved previously.

6.5 Regression Results

A fitting can be performed using the `Run fitting` button, which will attempt to run the Python code in the `Code` textbox while fitting the parameters under the `Fitting Parameters` popup using a Levenberg–Marquardt algorithm. If successful, the results will appear in the bottom half of the tab shown in Figure 6.7. The `Run fitting` button will only be active if a file has been loaded. If there is a syntax error in the code the fitting will not run at all. If there is another error in the code or the result variables `Zreal` and `Zimag` are not defined the fitting will result in an error.

Upon a successful fitting, a listbox will appear with all fitted parameters, their values, and their standard deviations and confidence intervals. If a parameter has a confidence interval greater than 100% it cannot be considered statistically different than 0 and it will be highlighted yellow.

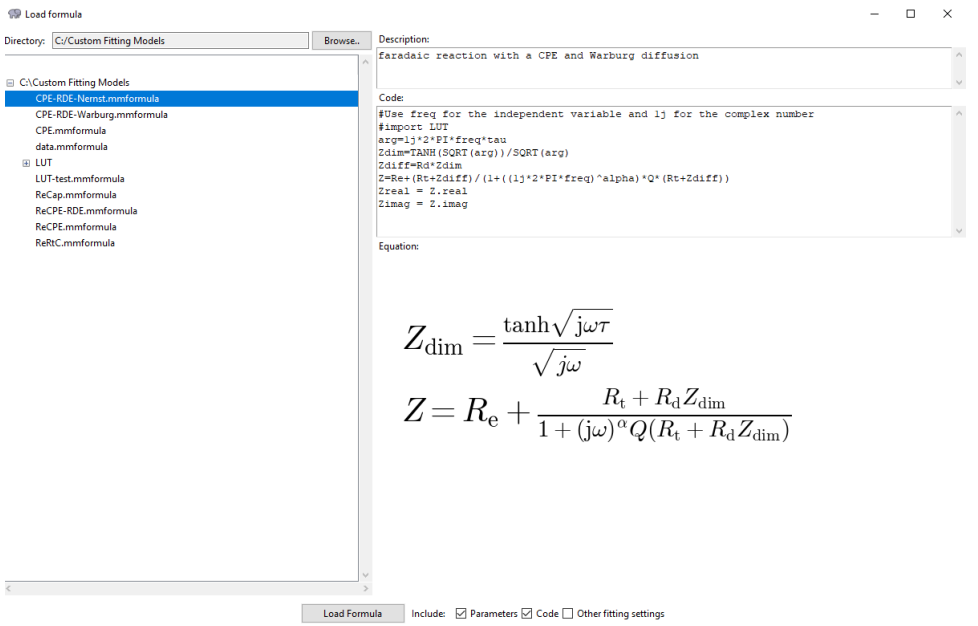

Figure 6.6: The Load Formula popup showing the file name, a description of the model, and the formula corresponding to the model. The default case for selection of formula will load the Python code and the associated variables and initial guesses. Selection of `Other fitting settings` will also load the number of Monte Carlo simulations, the weighting, and the fit type.

The `Advanced results` button will bring up a popup similar to the one discussed in Section 3.16, except that it does not include a correlation matrix. Values of parameter names, values, and standard deviations can be copied for pasting into a spreadsheet by clicking `Copy values and std. devs. as spreadsheet`. The `Plot` button and `Include Confidence Interval` checkbox act in the same manner as described in Section 3.19. The same plots are present, except for the R_e-adjusted plots and the derivative plot.

The residuals can be saved to a **.mmresiduals* file using the `Save Residuals` button. These can be used in the Error File Preparation tab; the only difference from those created in the Measurement Model Tab is that R_e is set as 0 in the file.

> **Warning!** The regression is very sensitive to initial guesses, to selection of the wrong model, and to typographical errors in the Python code.

6.6 Monte Carlo Standard Deviations

Under certain circumstances, the fitting may succeed but be unable to find the standard deviations of the results. In this case, a prompt will appear after the fitting terminates asking whether or not to perform Monte Carlo simulations in an attempt to estimate

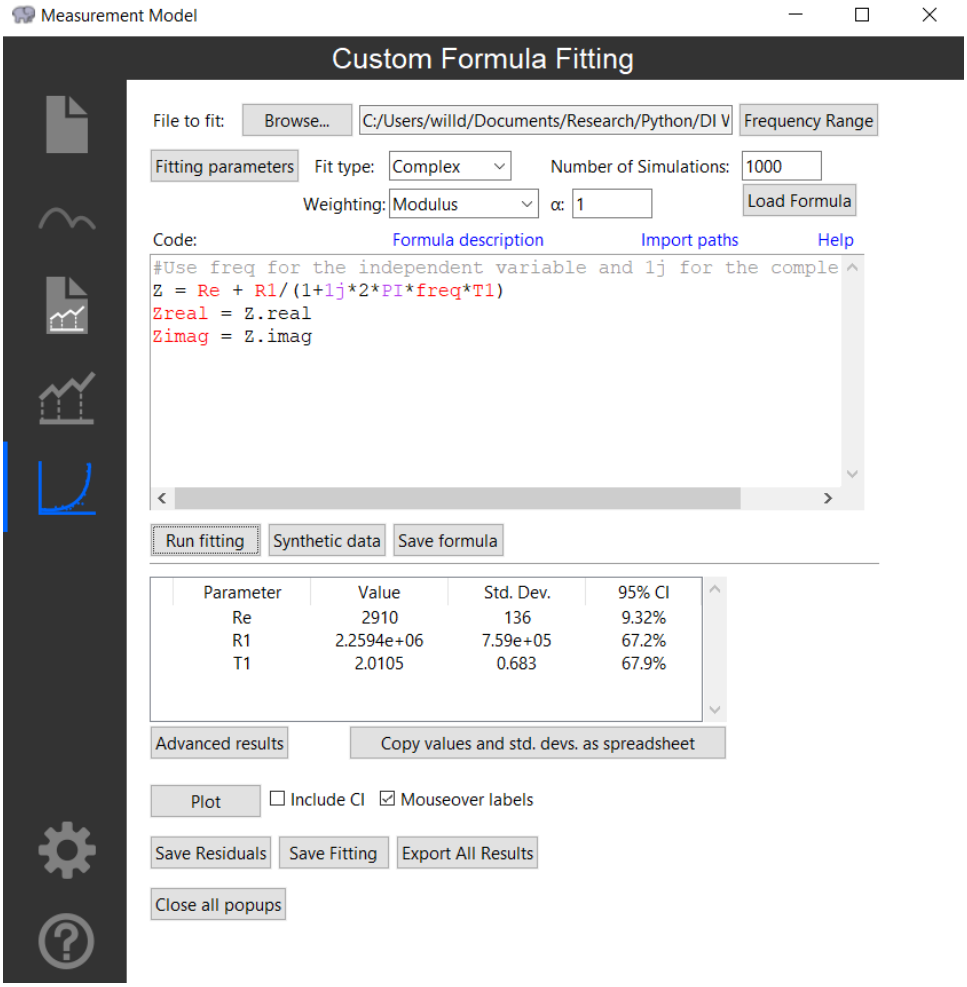

Figure 6.7: Dialog box showing a successful fitting of the custom model to the data. The value of parameters shown in Figure 6.2 are presented with the confidence interval.

the parameter standard deviations. If `No` is chosen, the fitting will finish as normal but with `nan` as each parameter's standard deviation. If `Yes` is chosen, another popup will appear asking for the number of Monte Carlo simulations to perform (see Figure 6.8).

Clicking `Cancel` will prevent any simulations from being performed (in the same manner as if `No` were chosen originally). Otherwise, `Run` can be clicked to perform the fits.

The program will then re-perform the fitting using the current code and original initial guesses. However, it will add normally distributed "noise" to the data centered at 0 with a variance equal to the standard deviation of the residuals of the initial fitting. After the fits are performed, the standard deviation between the different fitted values

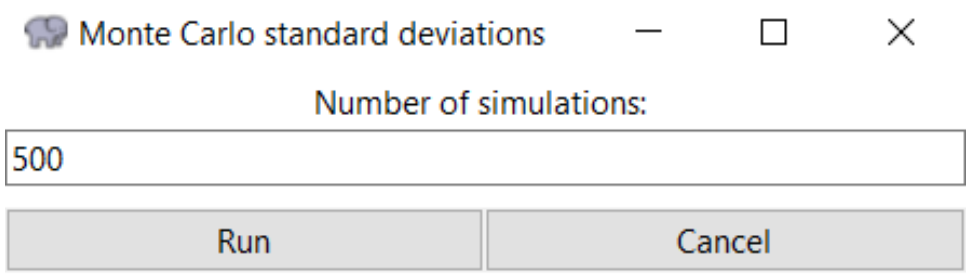

Figure 6.8: Choosing the number of Monte Carlo simulations for calculating the standard deviations for parameter estimates. Generally, a message that standard deviations of the results could not be calculated should be regarded as evidence of a problem with the regression.

of each parameter is used as that parameter's reported standard deviation.

During this process, a progress bar will appear in the main window showing what percentage of the fits have been performed. If more than 1000 simulations are chosen, multiple processes will be spawned to perform the fits. The fits can be cancelled as usual with the `Cancel fitting` button; in this case, no Monte Carlo results will be reported.

> **Tip!** Generally, a message that standard deviations of the results could not be calculated should be regarded as evidence of a problem with the regression.

6.7 Example

We fill fit a custom formula which will be for a single Voigt element in series with a resistance, i.e.,

$$Z = R_e + \frac{R_1}{1 + j\,(2\pi f)\,\tau_1} \tag{6.1}$$

1. First load a **.mmfile* under `Browse...`
2. Next, click `Fitting Parameters`, then click `Add Parameter` three times (one for each variable we will be fitting)
3. Rename `var0` to `Re`, `var1` to `R1`, and `var2` to `T1`, and type their initial guesses into the boxes to the right
4. In the dropdown to the right of `T1`, select `+` to constrain the time constant to be positive
5. Set the desired fit type, number of Monte Carlo simulations, and weighting strategy

6. Enter the following into the `Code` window (the code should syntax-highlight as seen below)

```
Z = Re + R1/(1+1j*2*PI*freq*T1)
Zreal = Z.real
Zimag = Z.imag
```

7. Click `Run fitting` and wait for results to appear

6.8 Synthetic Data

Synthetic impedance data can be generated by using the current code and parameter values with the `Synthetic data` button. The button will open a popup shown in Figure 6.9, where the frequencies can be specified at which data will be generated.

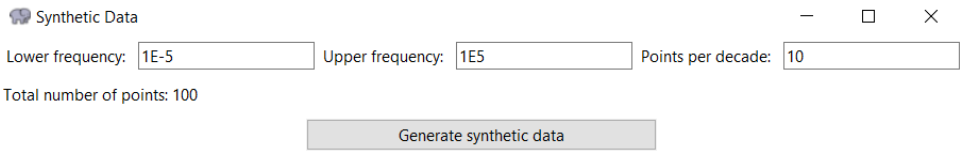

Figure 6.9: The synthetic data popup. This screen allows selection of the frequency range and number of logarithmically spaced frequencies per decade.

The leftmost textbox holds the lowest frequency, the middle box the uppermost frequency, and the rightmost box holds the number of data points per decade. The total of data points (calculated as the number of decades between the lower and upper frequency multiplied by the number of points per decade) is shown below.

After the choices are made, pressing the `Generate synthetic data` button will perform the calculations and display the results (see Figure 6.10). A listbox will appear with the frequencies to the left, the real impedance in the middle, and the imaginary impedance to the right. The data can be saved with te `Save` button as a measurement mode file (**.mmfile*), a text file (*.txt*), or a comma-separated values file (*.csv*). The data can also be copied in a format consistent with pasting to a spreadsheet program by using the `Copy values as spreadsheet` button. The `Plot` button will open a new window displaying plots of the synthetic data. These plots and the `Mousover labels` checkbox work in the same manner as in the Measurement Model tab (see Section 3.19), but without the residuals plots, the R_e-adjusted plots, or the derivative plot.

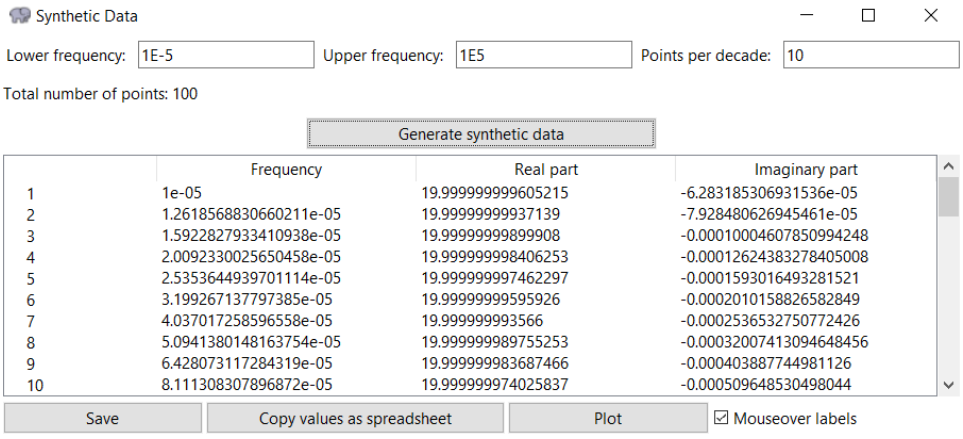

Figure 6.10: Synthetic data generated for an example formula consisting of one Voigt element with R_e=10 Ωcm^2, R=10 Ωcm^2, and $\tau = 10$ s. The synthetic data can be saved as a file, copied for a spreadsheet, and plotted.

6.9 Application to Sample Data

Three sample custom models are provided with this distribution. Of these, two are suitable for the sample data. Consider the file P120-1Q-33305s.mmfile with a maximum frequency of 3.8 kHz. Select the CPE-RDE-Nernst.mmformula model to fit the data. A description of the model is presented in Section 9.5.1. The result of the regression is presented in Figure 6.11 The Nernst plot shows some differences between the

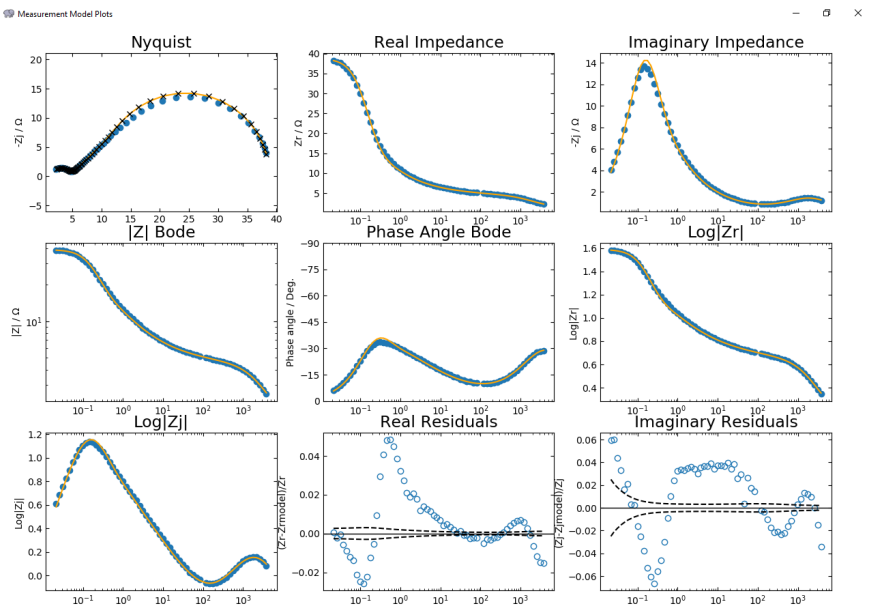

Figure 6.11: Comparison of the model to experimental data for the error-structure-weighted regression of the CPE-RDE-Nernst.mmformula to the file P120-1Q-14s.mmfile. Confidence interval for the data, i.e., $\pm 2\sigma$, are presented in the normalized residual error plots as black dashed lines. The χ^2/ν statistic had a value of 1787.

model and the data in mid-range frequencies, and the residual error plots show that the residual errors are much larger than the confidence interval based on the error structure of the data. The χ^2/ν statistic had a value of 1787, which is substantially larger than the expected value of unity.

The regression results are presented in Table 6.2. The CPE parameters may be expressed as a capacitance by use of the Brug formula,[45,46] e.g.,

$$C = Q^{1/\alpha} \left(R_e\right)^{(1-\alpha)/\alpha} \tag{6.2}$$

The calculated value $C = 18.7\ \mu\text{F/cm}^2$, presented in Table 6.2, is on the order of the value from the measurement model of $C = 24.7 \pm 2.9\ \mu\text{F/cm}^2$ presented in Table 3.6.

Select the ReCPE-RDE.mmformula model and use it to fit the data. The diffusion term for this model relies on the solution to the convective diffusion impedance for a

Table 6.2: Parameter values obtained by regression as shown in Figure 6.11.

Variable	Value	Units
R_e	1.244 ± 0.067	Ωcm^2
Q	80.4 ± 6.5	$\mu F/cm^2 s^{1-\alpha}$
R_d	34.170 ± 0.065	Ωcm^2
R_t	3.358 ± 0.074	Ωcm^2
α	0.863 ± 0.012	–
τ	2.515 ± 0.018	s
C	18.7	$\mu F/cm^2$

rotating disk electrode. No additional parameters are needed, but the model for the diffusion impedance should be more appropriate for the rotating disk electrode. The mathematical framework of the model is presented in Section 1.4.3. Take care to select the correct value for rotation speed in the model. The regression results are shown in Figure 6.12.The quality of the fit is much improved, and the χ^2/ν statistic had a value of 100.

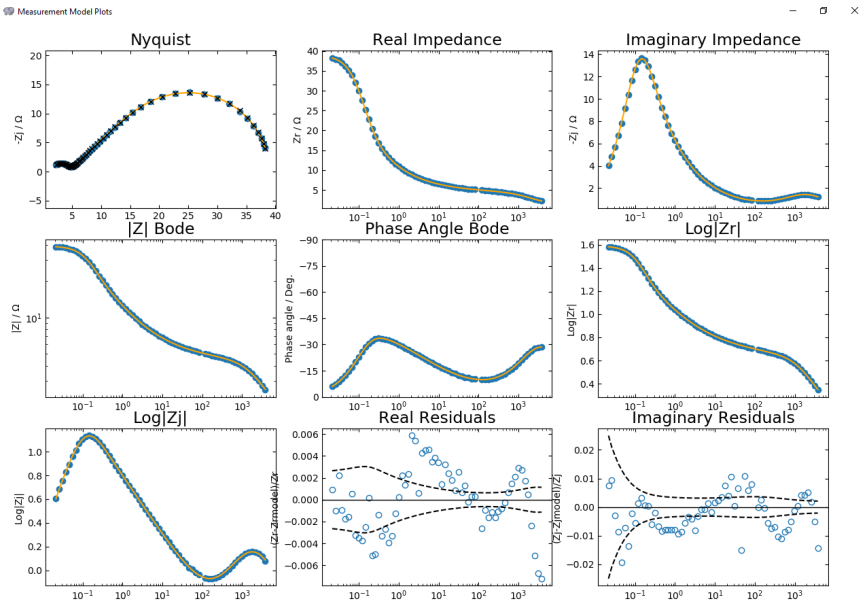

Figure 6.12: Comparison of the model to experimental data for the error-structure-weighted regression of the ReCPE-RDE.mmformula model to the file P120-1Q-14s.mmfile. Confidence interval for the data, i.e., $\pm 2\sigma$, are presented in the normalized residual error plots as black dashed lines. The χ^2/ν statistic had a value of 100.

The regression results are presented in Table 6.3. The Schmidt number is much larger than the expected value of 1000. This result may be attributed in part to use of a one-dimensional model to account for the convective diffusion impedance of a

Table 6.3: Parameter values obtained by regression as shown in Figure 6.12.

Variable	Value	Units
Sc	1235.0 ± 7.0	$-$
R_e	1.363 ± 0.014	Ωcm^2
Q	60.7 ± 1.2	$\mu F/cm^2 s^{1-\alpha}$
R_d	33.338 ± 0.016	Ωcm^2
R_t	3.173 ± 0.016	Ωcm^2
α	0.9014 ± 0.0029	$-$
C	21.7	$\mu F/cm^2$

rotating disk electrode under conditions for which current and potential distributions on the disk are not uniform. The capacitance extracted from equation (6.2),[45,46] given in Table 6.3 to be $C = 21.7\ \mu F/cm^2$, is close to the value from the measurement model as $C = 24.7 \pm 2.9\ \mu F/cm^2$ (see Table 3.6).

Chapter 7

Settings

Many of the defaults in the program can be changed in the settings tab, accessible through the gear icon(⚙). The settings are organized into five sub-tabs, each corresponding to a different part of the program, as shown in Figure 7.1.

7.1 Overall

The `Overall` tab includes settings that apply to the entire program. The `Light/Dark` toggle button will change the overall program theme (default: light). Light mode includes a white background and black text; dark mode includes a dark gray background and white text. The plot coloration will be changed as well. The `Side bar color` opens a color picker window where the color of the navigation and title panes can be set (default: dark gray). The tab title will change from white to black depending on its contrast with the color. The `Tab` dropdown is which tab the program opens to automatically (default: Input file). `Change tab on scroll`, if checked, will mean that the scrollwheel can be used to change tab so long as the mouse is over the navigation pane (default: checked).

7.2 Paths

The `Paths` tab allows selecting of file locations. `File directory` is the default directory for opening or saving files (default: C:\). `Formula directory` is the default directory for **.mmformula* files (default: C\). `Import paths` are the default extra paths searched by the Python interpreter for imports when performing a custom fit. The `Browse...` button here will add another directory to the listbox (instead of overwriting the current selection as the others do). To remove a directory from the listbox, select it and click `Remove` or right-click and select `Remove directory`.

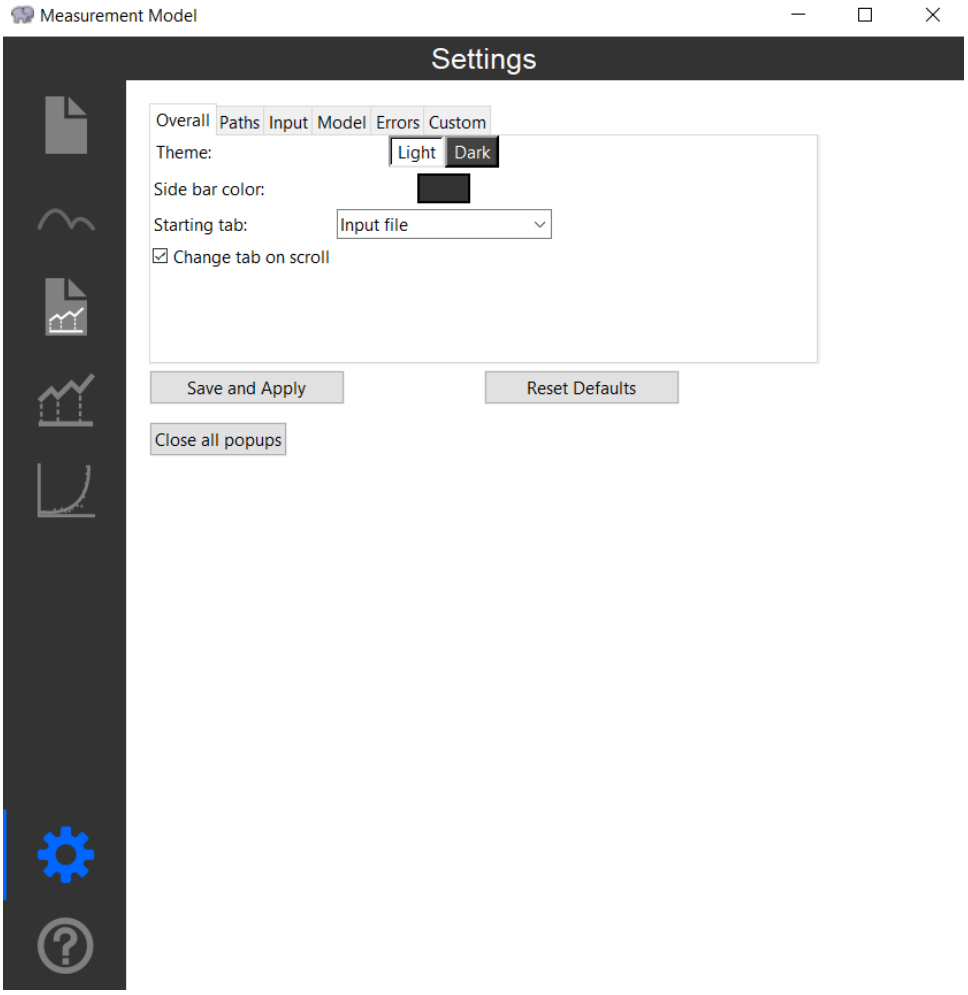

Figure 7.1: The Settings tab. Use this to change program defaults and to select input paths for data, custom models, and Python code used in the custom models.

7.3 Input

The `Detect number of comment lines` and `Detect delimiter` checkboxes make the program attempt to automatically detect the number of lines to ignore and the delimiter for data in the File Input and Conversion tab (default: checked). `Detect delimiter` can only be checked if `Detect number of comment lines` is checked. The textbox and dropdown beneath these two checkboxes is the default choice if automatic detection fails or is turned off (defaults: 0 and Tab, respectively). `Alert on close if unsaved` will - if checked - provide a confirmation alert if the program is closed while unsaved data is present in the File Input and Conversion tab (default: unchecked).

7.4 Model

`Number of simulations`, `Fit type`, `Weighting type`, and `Assumed noise level (α)` hold the default values for the fitting options described in Chapter 3 (defaults: 1000, Complex, Modulus, and 1, respectively). The `Error lines/ellipse color` button will open a color picker which will control the color of the confidence interval ellipses and lines when plotting in the Measurement Model or Custom Formula Fitting tabs (default: red). `Keep frequency range on loading new file` will, if checked, keep the number of points deleted when a new file is loaded (default: unchecked). `Undo frequency range with undo button` will, if checked, cause the `Undo Fit` button to remember the frequency range choices (default: unchecked).

7.5 Errors

The `Error Model Parameters` checkboxes control which parameters are selected by default (default: γ and δ). If `Re` is checked, it will check β as well; if β is unchecked, it will uncheck `Re`. The following three dropdowns control the default variance, moving average, and detrending options, respectively (defaults: Variance, 5 point, and Off, respectively).

7.6 Custom

The `Alert on close if unsaved` checkbox will - if checked - cause a confirmation prompt if a new custom formula has been entered, and the program is exited without saving it (default: unchecked). `Keep frequency range on loading new file` will - if checked - cause the frequency range choices (i.e., the number of points deleted) to remain the same when a new file is loaded (default: unchecked).

7.7 Saving and Resetting

The `Save and Apply` button will save the settings in a manner that will persist when the program is reloaded. It will also apply most appearance changes immediately (certain windows may need to be closed and re-opened for the changes to take effect). The `Reset Defaults` button will cause all settings to revert to their default state, and will apply those default settings.

Chapter 8

Help and About

The final tab is Help and About, which simply includes links to libraries used in the program, as well as contact information. Also included is a brief copyright notice that the program is distributed under the GNU General Public License. The license can be found online at https://www.gnu.org/licenses. A copy of the license is distributed with the program as an HTML file; it can be opened by clicking the `GNU General Public License` link at the bottom of the tab. The tab can be accessed either by clicking the question mark icon in the navigation pane () or by pressing `F1` from anywhere in the program.

8.1 Modules

Many modules/libraries were used in creating this program. These include:

- The standard library
- Delimiter detection: detect_delimiter
- COM: comtypes
- Copying to/from clipboard: pyperclip
- File detection: impedance.py
- Fitting: LMFIT
- Graphing: matplotlib
- Images: PIL (Pillow)
- MPR decoding: galvani
- Numerical manipulations: numpy
- Other fitting: scipy

8.2 Contact

This section includes contact information for the authors.

Prof. Mark E. Orazem	(352) 392-6207
	morazem@che.ufl.edu
William Watson	williamdwatson1@gmail.com

Chapter 9

Procedure

The procedure for analyzing data is presented here in brief outline form.

9.1 Preliminary Analysis for Error Structure

1. Convert all data files to **.mmfile*s using the File Tools and Conversion tab (Chapter 2).
2. Perform complex fits for all replicate data using the Measurement Model tab (Chapter 3).
3. Obtain the maximum number of line shapes for each replicate.
4. Save the residual errors by using the `Save Residuals for Error Analysis` button. Note that, for a given set of replicates, the number of line shapes (Voigt elements) for each file needs to be the same in order to obtain the error structure. In addition, the number of line shapes for all files must be equal to the number of line shapes of the file with the minimum number of line shapes. For example if there are 5 files for a given set of replicates and 6 line shapes were obtained for 4 files and 5 line shapes were obtained for the remaining file, the files with six line shapes need to be reanalyzed with five line shapes. This can be done by repeating the regressions until five line shapes have been obtained.
5. Compile the residual errors using the Error File Preparation tab (Chapter 4).
6. Use the Error Analysis tab (Chapter 5) to calculate parameters for the error structure model replicates.

9.2 Assess Consistency with the Kramers–Kronig Relations

There are several ways to assess consistency with the Kramers–Kronig relations. In principle, since the Voigt model is itself consistent with the Kramers–Kronig relations,

the ability to fit this model to data within the noise level of the measurement should indicate that the data are consistent. An ambiguity exists when the data are not fully consistent because the lack of fit of the model could be due to causes other than inconsistency with the Kramers–Kronig relations. Some other possible causes could be that the number of frequencies measured was insufficient to allow regression with a large enough number of Voigt parameters or the initial guesses for the non-linear regression could be poorly chosen.

Regression to one component with subsequent prediction of the other component provides a more sensitive method to assess consistency. A procedure for this analysis is described below:

1. Perform a fit to the imaginary part of the spectrum using error structure weighting. Increase the number of lineshapes used until the maximum number of statistically significant parameters is obtained.

2. Examine the imaginary residual errors to determine if they fall within the error structure. Should a few points lie outside the error structure at intermittent frequency values do not be concerned. Assess prediction of real part of the impedance by examining real residual plots with confidence intervals displayed. Real residual data points that are outside the confidence interval are considered to be inconsistent with the Kramers–Kronig relations and should be removed from the data set.

3. Typically, the number of lineshapes that can be determined in a complex fit will increase when Kramers–Kronig -inconsistent data are removed. Deletion of data that are strongly influenced by bias errors increases the amount of information that can be extracted from the data. In other words, the bias in the complete data set induces correlation in the model parameters which reduces the number of parameters that can be identified. Removal of the biased data results in a better conditioned data set that enables reliable identification of a larger set of parameters.

Tip! R_e cannot be obtained by fitting the Voigt model to the imaginary part of the spectrum. For imaginary fits, R_e should be treated as an adjustable parameter. The `Update Re` button in the Measurement Model tab - described in Section 3.20 - provides a convenient way to manually adjust the value of the solution resistance to determine whether a value can be found that would bring all the real part of the impedance into compliance with the Kramers–Kronig relations.

The use of measurement models is superior to the use of polynomial fitting because fewer parameters are needed to model complex behavior, and because the measurement model satisfies the Kramers–Kronig relations implicitly. Experimental data

can, therefore, be checked for consistency with the Kramers–Kronig relations without actually integrating the equations over frequency, avoiding the concomitant quadrature errors. The use of measurement models does require an implicit extrapolation of the experimental data set, but the implications of the extrapolation procedure are quite different from extrapolations reported in the literature. The extrapolations done with measurement models are based on a common set of parameters for the real and imaginary parts and on a model structure that has been shown to represent adequately the observations. The confidence in the extrapolation using measurement models is, therefore, higher. For the application to a preliminary screening of the data, the use of measurement models is superior to the use of more specific electrical circuit analogues because one can determine whether the residual errors are due to an inadequate model, to failure of data to conform to the Kramers–Kronig assumptions, or to experimental noise. The algorithm proposed in this work, in conjunction with data-based weighting, provides a robust way to check for consistency of impedance data.

It should be emphasized that the work presented here is part of an overall assessment of measurement errors. The measurement model is used as a filter for lack of replication to obtain a quantitative value for the standard deviation of the measurement as a function of frequency. The mean error identified in this way is equal to zero, thus the standard deviation of the measurement does not incorporate the bias errors. In contrast, the standard deviation of repeated impedance measurements typically includes a significant contribution from bias errors because perfectly replicate measurements can rarely be made for electrochemical systems. Since the line-shapes of the measurement model satisfy the Kramers–Kronig relations, the Kramers–Kronig relations can then be used as a statistical observer to assess the bias error in the measurement.

9.3 Preliminary Modeling of Impedance Data

As used here, the measurement model provides much more than a preliminary analysis of impedance data in terms of the number of resolvable time constants and asymptotic values. The measurement model can be used as a filter for lack of replication that allows accurate assessment of the standard deviation of impedance measurements. This information is critical for selection of weighting strategies for regression, provides a quantitative basis for assessment of the quality of fits, and can guide experimental design. The measurement model is also used to assess the bias component of the data. The next step in the interpretation of these data is the development of deterministic models which can account for the physical phenomena associated with this system. The analysis presented here can be used to ensure that the data used for comparison to the model are not corrupted by bias errors, thus facilitating interpretation in terms of physical parameters.

A preliminary interpretation of the impedance data can be obtained by using the `Evaluate Simple Parameters` button (see Section 3.18). The procedure is:

1. Perform a complex fit to the spectrum using error structure weighting and deleting data points found to be inconsistent with the Kramers–Kronig relations. Increase the number of lineshapes used until the maximum number of statistically significant parameters is obtained.

2. Press the `Evaluate Simple Parameters` button.

The time constants and resistances obtained provide a preliminary analysis of the impedance data that can guide development of process models as discussed in Chapter 6.

9.4 Estimation of Capacitance

The measurement model also serves as a useful means to provide quantitative estimates for parameters relevant to impedance spectroscopy data, including ohmic resistance, polarization resistance, and capacitance. The following discussion is taken from Liao et al.[4]

The ability to extract capacitance by use of the measurement model is important, because most impedance systems show distributed-time-constant behavior that is often fit by use of the CPE. The CPE parameters are related to the capacitance of the electrode, but extraction of a capacitance requires different approaches depending on the nature of the underlying distribution. For example, the Brug formula[45] is used in cases of a surface distribution of time constants, and the power-law model[23,24] applies for a distribution through a film. Not all distributed-time-constant behaviors are represented by a CPE, and formulas like the Brug and Power-Law equations do not exist for such systems. Furthermore, the measurement model provides a unique ability to extract the high-frequency and low-frequency ohmic resistances for systems exhibiting an ohmic impedance.

The application of the measurement model to extract parameters for experimental impedance data depends on the nature of the electrochemical system under investigation. For systems that have a uniform current and potential on the electrode surface, the ohmic resistance and capacitance may be extracted by regression of the measurement model to the part of the impedance spectrum that is found to be consistent with the Kramers–Kronig relations. The polarization resistance corresponding to the zero-frequency limit may also be estimated. For systems that have a nonuniform current and potential distribution, a characteristic frequency can be identified above which the impedance is influenced by the associated frequency dispersion.[31,32] This characteristic frequency may be estimated following equation (3.5) where values of C and $R_{e,HF}$ are estimated by regression of the measurement model to the impedance spectrum that is found to be consistent with the Kramers–Kronig relations. The estimate for C may be updated be eliminating the frequencies above f_c. The new value of C is used to calculate a new estimate for the characteristic frequency. This iterative process

is completed when the estimated characteristic frequency is larger than the maximum frequency used in the regression. The iterative process will yield estimates for C and $R_{e,LF}$. The iterative process may fail to yield a suitable convergence for cases in which $R_t < R_{e,HF}$.

The use of the measurement model to extract physical properties of electrochemical systems are guided by four principles:

1. In the presence of geometry-induced frequency dispersion, the capacitance obtained by the measurement model is smaller than the input double-layer capacitance. The measurement model can be made to yield an accurate capacitance by removing the contribution of the ohmic impedance from the regressed data.

2. High-frequency and low-frequency ohmic resistances may be defined for systems showing the presence of geometry-induced frequency dispersion. The low-frequency ohmic resistance may be quantified by regression to data that are below the characteristic frequency defined by equation (3.5).

3. For systems that exhibit the influence of geometric capacitance, the capacitance obtained by the measurement model is the geometric capacitance, independent of the value of the double layer capacitance. The double-layer capacitance and low-frequency ohmic resistance may be extracted from regression to data that are below the characteristic frequency defined by equation (3.5). The high-frequency ohmic resistance may be obtained by regression to data that are at frequencies below that affected by the geometric capacitance loop.

4. For systems that show pure constant-phase-element behavior, a capacitance cannot be identified, neither by the measurement model nor by extrapolation of the Cole-Cole plot.

Used wisely, the measurement model may be a powerful tool for extracting the capacitance, ohmic resistance, and polarization resistance for impedance data. It can be applied both for systems exhibiting normal distributions of time constants and, if applied to frequencies sufficiently below the characteristic frequency for geometry-induced dispersion, for systems showing surface distributions of time-constants. Used in this way, the measurement model provides a powerful complement to the development of system-specific process models.

9.5 Application of Custom Models

It will be convenient to create a catalog of custom process models. Some examples are presented here that could help the user begin to create a catalog.

9.5.1 Faradaic Reaction with a CPE and Finite Film Diffusion

The dimensionless diffusion impedance for a film can be expressed as

$$Z_{\text{dim}} = \frac{\tanh\sqrt{\text{j}\omega\tau}}{\sqrt{j\omega}} \tag{9.1}$$

and the impedance for a faradaic reaction influenced by diffusion and coupled with a constant-phase-element behavior is given as

$$Z = R_{\text{e}} + \frac{R_{\text{t}} + R_{\text{d}}Z_{\text{dim}}}{1 + (\text{j}\omega)^{\alpha}Q(R_{\text{t}} + R_{\text{d}}Z_{\text{dim}})} \tag{9.2}$$

The corresponding entry in the code textbox, as discussed in Section 6.3, may be written as

```
arg=1j*2*PI*freq*tau
Zdim=TANH(SQRT(arg))/SQRT(arg)
Zdiff=Rd*Zdim
Z=Re+(Rt+Zdiff)/(1+((1j*2*PI*freq)^alpha)*Q*(Rt+Zdiff))
Zreal = Z.real
Zimag = Z.imag
```

where the fitting parameters are `Re`, `Q`, `Rd`, `Rt`, `alpha`, and `tau`.

9.5.2 Faradaic Reaction with a CPE and Warburg Diffusion

The diffusion impedance for an infinite film is attributed to Warburg.[47,48] The corresponding dimensionless diffusion impedance is given by

$$Z_{\text{dim}} = \frac{1}{\sqrt{\text{j}\omega\tau}} \tag{9.3}$$

The diffusion resistance R_{d} used in equation (9.2) cannot be used directly for a Warburg impedance because the parameter τ and R_{d} are not independent. Thus, a modified parameter is used, i.e.,

$$R_{\text{d}}^{*} = R_{\text{d}}/\sqrt{\tau} \tag{9.4}$$

and the governing equation, given as

$$Z = R_{\text{e}} + \frac{R_{\text{t}} + R_{\text{d}}^{*}Z_{\text{dim}}}{1 + (\text{j}\omega)^{\alpha}Q(R_{\text{t}} + R_{\text{d}}^{*}Z_{\text{dim}})} \tag{9.5}$$

is the same as equation (9.2) except that R_{d} is replaced by R_{d}^{*} and Z_{dim} is defined by equation (9.3).

The corresponding entry in the code textbox, as discussed in Section 6.3, may be written as

```
arg=1j*2*PI*freq
Zdim=1/SQRT(arg)
Zdiff=Rd*Zdim
Z=Re+(Rt+Zdiff)/(1+((1j*2*PI*freq)^alpha)*Q*(Rt+Zdiff))
Zreal = Z.real
Zimag = Z.imag
```

where the fitting parameters are `Re`, `Q`, `Rd`, `Rt`, and `alpha`. The examples shown in Section 9.5.1 aand the present section highlight the need to ensure that all parameters used in the regression are independent.

Chapter 10

Troubleshooting

From time to time, errors may arise. Errors generated by the program will most often include a number which can be referenced to Table 10.1.

Table 10.1: Error Codes.

Error code	Corrective Action
2	When opening multiple files with the program, all extensions must be **.mmresiduals* or **.mmerrors* (exclusive or, no mixing)
3	There was an error opening the files (try opening them from the program when the program itself is running)
5	There was an error opening files (the file may be corrupted)
6	Number of values for frequency, real impedance, and imaginary impedance don't match; try manually editing file to remove bad points
7	Multiplier value is invalid; ensure that the multiplier is a real number
8	Column number is missing; ensure that the column numbers are all present, and are all positive integers
9	Problem reading the input file; don't manually edit **.mmfitting* files
10	File has an unknown extension; only **.mmfile* or **.mmfitting* files will work
11	The fitting failed (no minimization of the objective function could be found); try multistart/different initial guesses or a different number of parameters
12	A minimization was found, but parameter standard errors couldn't be estimated; try fitting again, or try multistart/different initial guesses/a different number of parameters
18	R_e's initial guess is negative but it is constrained positive; remove the initial guess or change the constraint

Table 10.1: (continued)

Error code	Corrective Action
19	One or more initial guesses are missing; ensure that all parameters have real-number initial guesses under `Edit Model Parameters`
20	One or more parameters has an initial guess whose sign does not match its constraint; change the initial guess or change the constraint
21	One or more of the initial guesses has an invalid value; initial guesses must be a real number
22	No parameters were chosen for error structure; choose at least one parameter
23	Error stucture has an invalid value; all values must be real numbers, even if not checked
24	All error structure parameters are 0 (similar to 22); make at least one parameter non-zero
25	The assumed noise (α) value is invalid; it must be a real number
26	Input file has an unknown extension; all files must have a *.*mmresiduals* extension
27	Number of data is different between files; all input files must have the same number of data, so try manually editing the files or refitting under Measurement Model with the same number of data chosen using `Frequency Range`
28	Frequencies do not match; all frequencies must be the same between files - correct as in 27 or re-do measurements
29	Error reading file; the file may be corrupted
30	Input file has an unknown extension; all files must have a *.*mmerrors* extension
31	Number of data do not match within a *.*mmerrors* file; do not manually edit *.*mmerrors* files
33	Error reading file; the file may be corrupted
34	Fitting failed as a minimization of the objective function could not be found; try changing the error structure or using different initial guesses
35	Too few values for a 3-point moving average; at least 3 points are needed
36	Too few values for a 5-point moving average; at least 5 points are needed
37	Initial guess for α is invalid; the guess must be a real number
38	Initial guess for β is invalid; the guess must be a real number

Table 10.1: (continued)

Error code	Corrective Action
39	Initial guess for γ is invalid; the guess must be a real number
40	Initial guess for δ is invalid; the guess must be a real number
41	No parameters are chosen for fitting; choose at least one parameter
42	Error reading file; the file may be corrupted
43	Input file has an unknown extensions; it must have a **.mmfile* or **.mmcustom* extension
44	No formula is present; input a valid formula
45	A variable name is a Python reserved word; see online for a list of Python reserved words, and change the offending variable's name
47	A variable name is a word used in the fitting code (i.e., `Zr`, `Zj`, `freq`, `Zreal`, `Zimag`, or `weighting`); change the offending variable's name
48	A variable name is used more than once; ensure all variables have a unique name
49	Number of Monte Carlo simulations is invalid; ensure the `Number of Simulations` textbox holds a positive integer
50	Assumed noise (α) is invalid; ensure it is a real number
51	Number of comment lines is invalid; ensure it is a positive integer
52	Error applying or saving settings; ensure all values are valid, try saving again, or try restarting the program; do not move or manually edit the *settings.ini* file
53	The **.mmfile* linked in the loaded **.mmcustom* file could not be located; check that the file has not been moved, renamed, or deleted
54	One or more initial guesses is invalid; change the offending variable(s) value to a real number
55	The lower bound for multistart is invalid; change the lower bound to a real number
56	The upper bound for multistart is invalid; change the upper bound to a real number
57	The number of multistarts is invalid; change the number of multistarts to a positive integer
58	The signs of the upper and lower bounds when using logarithmic spacing are different; change the signs so that they match, or use a different spacing
59	Either the upper or lower bound is zero when using logarithmic spacing; change the bound to be non-zero

Table 10.1: (continued)

Error code	Corrective Action
60	The custom multistart choices couldn't be parsed; ensure that all custom multistart values are real numbers only, and that they are all separated by a comma
61	Either the upper or lower limit of a parameter is not a real number; change the offending limit to a real number
62	The formula that was clicked could not be loaded; do not manually edit a *.mmformula file
63	Specified import path does not exist; change the offending import path so it points to an existing directory
64	There is an invalid value for either the upper or lower frequency, or the number of simulated points per decade; ensure that the upper and lower frequencies are real numbers, and that the number of points per decade is a positive integer
65	The total number of simulated points is negative; ensure that the upper frequency is greater than the lower frequency and that the number of points per decade is a positive integer

References

1. P. Agarwal, M. E. Orazem, and L. H. García-Rubio, "Measurement Models for Electrochemical Impedance Spectroscopy: 1. Demonstration of Applicability," *Journal of the Electrochemical Society*, **139** (1992) 1917–1927.
2. P. Agarwal, O. D. Crisalle, M. E. Orazem, and L. H. García-Rubio, "Measurement Models for Electrochemical Impedance Spectroscopy: 2. Determination of the Stochastic Contribution to the Error Structure," *Journal of the Electrochemical Society*, **142** (1995) 4149–4158.
3. P. Agarwal, M. E. Orazem, and L. H. García-Rubio, "Measurement Models for Electrochemical Impedance Spectroscopy: 3. Evaluation of Consistency with the Kramers-Kronig Relations," *Journal of the Electrochemical Society*, **142** (1995) 4159–4168.
4. H. Liao, W. Watson, A. Dizon, B. Tribollet, V. Vivier, and M. E. Orazem, "Physical Properties Obtained from Measurement Model Analysis of Impedance Measurements," *Electrochimica Acta*, **354** (2020) 136747.
5. M. E. Orazem, N. Pébère, and B. Tribollet, "Enhanced Graphical Representation of Electrochemical Impedance Data," *Journal of the Electrochemical Society*, **153** (2006) B129–B136.
6. M. E. Orazem and B. Tribollet, *Electrochemical Impedance Spectroscopy*, 2nd edition (Hoboken: John Wiley & Sons, 2017).
7. L. H. García-Rubio and N. Ro, "Detailed Copolymer Characterization Using Ultraviolet Spectroscopy," *Canadian Journal of Chemistry*, **63** (1985) 253–263.
8. L. H. García-Rubio, "The Effect of Molecular Size on the Absorbtion Spectra of Macromolecules," *Macromolecules*, **20** (1987) 3070–3075.
9. L. H. García-Rubio, C. Lopez-Menacho, and S. Grossman, "Characterization of Proteins During Aggregation II: Use of Model Molecules for Spectroscopy Analysis," *Chemical Engineering Communications*, **122** (1993) 85–101.
10. P. Agarwal, M. E. Orazem, and L. H. García-Rubio, "Application of the Kramers Kronig Relations to Electrochemical Impedance Spectroscopy," in *Electrochemical Impedance: Analysis and Interpretation*, J. Scully, D. Silverman, and M. Kendig, edi-

tors, volume ASTM STP 1188 (Philadelphia: American Society for Testing and Materials, 1993) 115–139.

11. M. Durbha, M. E. Orazem, and L. H. García-Rubio, "Spectroscopy Applications of the Kramers-Kronig Relations: Implications for Error Structure Identification," *Journal of the Electrochemical Society*, **144** (1997) 48–55.
12. S. L. Carson, M. E. Orazem, O. D. Crisalle, and L. H. García-Rubio, "On the Error Structure of Impedance Measurements: Simulation of Frequency Response Analysis (FRA) Instrumentation," *Journal of the Electrochemical Society*, **150** (2003) E477–E490.
13. S. L. Carson, M. E. Orazem, O. D. Crisalle, and L. H. García-Rubio, "On the Error Structure of Impedance Measurements: Simulation of Phase Sensitive Detection (PSD) Instrumentation," *Journal of the Electrochemical Society*, **150** (2003) E491–E500.
14. P. K. Shukla, M. E. Orazem, and O. D. Crisalle, "Validation of the Measurement Model Concept for Error Structure Identification," *Electrochimica Acta*, **49** (2004) 2881–2889.
15. L. Pauwels, W. Simons, A. Hubin, J. Schoukens, and R. Pintelon, "Key Issues for Reproducible Impedance Measurements and Their Wellfounded Error Analysis in a Silver Electrodeposition System," *Electrochimica Acta*, **47** (2002) 2135–2141.
16. O. Blajiev, T. Breugelmans, R. Pintelon, and A. Hubin, "Improvement of the Impedance Measurement Reliability by Some New Experimental and Data Treatment Procedures applied to the Behavior of Copper in Neutral Chloride Solutions containing Small Heterocycle Molecules," *Electrochimica Acta*, **51** (2006) 1403–1412.
17. B. A. Boukamp, "A Linear Kronig-Kramers Transform Test for Immittance Data Validation," *Journal of the Electrochemical Society*, **142** (1995) 1885–1894.
18. C. You, M. A. Zabara, M. E. Orazem, and B. Ulgut, "Application of the Kramers–Kronig Relations to Multi-Sine Electrochemical Impedance Measurements," *Journal of The Electrochemical Society*, **167** (2020) 020515.
19. M. E. Orazem, P. T. Wojcik, M. Durbha, I. Frateur, and L. H. García-Rubio, "Application of Measurement Models for Interpretation of Impedance Spectra for Corrosion," *Materials Science Forum*, **289-292** (1998) 813–828.
20. C. Cleveland, S. Moghaddam, and M. E. Orazem, "Nanometer-Scale Corrosion of Copper in Deaerated Deionized Water," *Journal of the Electrochemical Society*, **161** (2014) C107–C114.
21. M. E. Orazem, P. K. Shukla, and M. A. Membrino, "Extension of the Measurement Model Approach for Deconvolution of Underlying Distributions for Impedance Measurements," *Electrochimica Acta*, **47** (2002) 2027–2034.
22. Y.-M. Chen, A. S. Nguyen, M. E. Orazem, B. Tribollet, N. Pébère, M. Musiani, and V. Vivier, "Identification of Resistivity Distributions in Dielectric Layers by Measurement Model Analysis of Impedance Spectroscopy," *Electrochimica Acta*, **219** (2016) 312–320.

23. B. Hirschorn, M. E. Orazem, B. Tribollet, V. Vivier, I. Frateur, and M. Musiani, "Constant-Phase-Element Behavior Caused by Resistivity Distributions in Films: 1. Theory," *Journal of the Electrochemical Society*, **157** (2010) C452–C457.
24. B. Hirschorn, M. E. Orazem, B. Tribollet, V. Vivier, I. Frateur, and M. Musiani, "Constant-Phase-Element Behavior Caused by Resistivity Distributions in Films: 2. Applications," *Journal of the Electrochemical Society*, **157** (2010) C458–C463.
25. C. You, A. Titov, B. H. Kim, and M. E. Orazem, "Impedance Measurements on QLED Devices: Analysis of High-Frequency Loop in Terms of Material Properties," *Journal of Solid State Electrochemistry*, (2020) submitted.
26. M. E. Orazem, "Electrochemical Impedance Spectroscopy: The Journey to Physical Understanding," *Journal of Solid State Electrochemistry*, (2020).
27. M. E. Orazem, M. Durbha, C. Deslouis, H. Takenouti, and B. Tribollet, "Influence of Surface Phenomena on the Impedance Response of a Rotating Disk Electrode," *Electrochimica Acta*, **44** (1999) 4403–4412.
28. M. Durbha, M. E. Orazem, and B. Tribollet, "A Mathematical Model for the Radially Dependent Impedance of a Rotating Disk Electrode," *Journal of the Electrochemical Society*, **146** (1999) 2199–2208.
29. B. Tribollet and J. S. Newman, "The Modulated Flow at a Rotating Disk Electrode," *Journal of the Electrochemical Society*, **130** (1983) 2016–2026.
30. J. S. Newman, "Frequency Dispersion in Capacity Measurements at a Disk Electrode," *Journal of the Electrochemical Society*, **117** (1970) 198–203.
31. V. M.-W. Huang, V. Vivier, M. E. Orazem, N. Pébère, and B. Tribollet, "The Apparent CPE Behavior of an Ideally Polarized Blocking Electrode: A Global and Local Impedance Analysis," *Journal of the Electrochemical Society*, **154** (2007) C81–C88.
32. O. Gharbi, A. Dizon, M. E. Orazem, M. T. T. Tran, B. Tribollet, and V. Vivier, "From Frequency Dispersion to Ohmic Impedance: A New Insight on The High-Frequency Impedance Analysis of Electrochemical Systems," *Electrochimica Acta*, **320** (2019) 134609.
33. A. R. Dizon and M. E. Orazem, "On the Impedance Response of Interdigitated Electrodes," *Electrochimica Acta*, **327** (2019) 135000.
34. P. Agarwal, O. C. Moghissi, M. E. Orazem, and L. H. García-Rubio, "Application of Measurement Models for Analysis of Impedance Spectra," *Corrosion*, **49** (1993) 278–289.
35. M. E. Orazem, P. Agarwal, C. Deslouis, and B. Tribollet, "Application of Measurement Models to Electrohydrodynamic Impedance Spectroscopy," *Journal of the Electrochemical Society*, **143** (1996) 948–960.
36. M. E. Orazem, T. E. Moustafid, C. Deslouis, and B. Tribollet, "The Error Structure of Impedance Spectroscopy Measurements for Systems with a Large Ohmic Resistance with Respect to the Polarization Impedance," *Journal of the Electrochemical Society*, **143** (1996) 3880–3890.

37. M. A. Membrino, M. E. Orazem, E. Scott, and J. B. Phipps, "Electrochemical Impedance Measurements For Characterization of Ion Transport Pathways In Skin," in *Minutes: Transdermal Administration: A Case Study, Iontophoresis* (Paris, France: Editions de Santé, 1997) 313.

38. I. Frateur, C. Deslouis, M. E. Orazem, and B. Tribollet, "Modeling of the Cast Iron/-Drinking Water System by Electrochemical Impedance Spectroscopy," *Electrochimica Acta*, **44** (1999) 4345–4356.

39. M. E. Orazem, P. Agarwal, A. N. Jansen, P. T. Wojcik, and L. H. García-Rubio, "Development of Physico-Chemical Models for Electrochemical Impedance Spectroscopy," *Electrochimica Acta*, **38** (1993) 1903–1911.

40. J. R. Dygas and M. W. Breiter, "Variance of Errors and Elimination of Outliers in the Least Squares Analysis of Impedance Spectra," *Electrochimica Acta*, **44** (1999) 4163–4174.

41. J. R. Macdonald, "Some New Directions in Impedance Spectroscopy Data Analysis," *Electrochimica Acta*, **38** (1993) 1883–1890.

42. J. R. Macdonald and V. I. Piterbarg, "On the Transformation of Colored Random Noise by the Kronig-Kramers Integral Transforms," *Journal of Electroanalytical Chemistry*, **428** (1997) 1–9.

43. S. L. Carson, *Application of Complex Spectroscopic Techniques to the Characterization of particles in Suspension*, Ph.D. dissertation, University of Florida (1999).

44. S. L. Carson, M. E. Orazem, O. D. Crisalle, and L. H. García-Rubio, "Influence of Instrumentation on the Error Structure of Impedance Measurement," in *Electrochemical Approach to Selected Corrosion and Corrosion Control Studies*, P. L. Bonora and F. Deflorian, editors (London: The Institute of Materials, 2000) 344–360.

45. G. J. Brug, A. L. G. van den Eeden, M. Sluyters-Rehbach, and J. H. Sluyters, "The Analysis of Electrode Impedances Complicated by the Presence of a Constant Phase Element," *Journal of Electroanalytical Chemistry*, **176** (1984) 275–295.

46. M. E. Orazem, B. Tribollet, V. Vivier, S. Marcelin, N. Pébère, A. L. Bunge, E. A. White, D. P. Riemer, I. Frateur, and M. Musiani, "Dielectric Properties of Materials Showing Constant-Phase-Element (CPE) Impedance Response," *Journal of the Electrochemical Society*, **160** (2013) C215–C225.

47. E. Warburg, "Über das Verhalten sogenannter unpolarisirbarer Elektroden gegen Wechselstrom," *Annalen der Physik und Chemie*, **67** (1899) 493–499.

48. E. Warburg, "Über die Polarisationscapacität des Platins," *Annals of Physics*, **6** (1901) 125–135.

Author Index

The references for this book are listed on page 105 in numerical order of appearance. The corresponding authors of the works are presented in this index. The numbers refer to the cited works given on page 105.

Index

Numbers followed by an “t” refer to the page where the corresponding entry can be found in a table.