ChiSurf

An incomplete guide to global analysis

25.02.09

Contents

1. INTRODUCTION 3

1.1. Overview 3
1.2. Data import 5
1.3. Fit models 8
1.4. Creating fits 8
1.5. Analysis dock 9
1.6. Parameters 9
1.7. Fit interface 11
1.8. Model scores 12
1.9. Model scoring range 12
1.10. Parameter optimization 13
1.11. Parameter sampling 15
1.12. Multiple document interface 16
1.13. Fit Plots 17
1.14. Parameter scan 17

2. EXPERIMENTS 18

2.1. Time resolved fluorescence 18
2.2. Fluorescence correlation spectroscopy 24
2.3. Global models 24

3. SCRIPTS 24

3.1. FRET lines 24

4. SETTING FILES 24

5. TOOLS 24

5.1. CLSMView 25
5.2. K2distribution 25
5.3. F-Test calculator 25

6. PLUGINS & TOOLS 25

6.1. nDxplorer 25
6.2. CLSMview 25
6.3. Correlation 28
6.4. Global view 32

7. TUTORIALS 35

7.1. Analysis of live-cell FCCS experiments 35
7.2. Analysis of Calibration measurements 36
7.3. Analysis of live cell experiments 56
7.4. Analysis of simulated data 62
7.5. References 68

1. Introduction

1.1. Overview

ChiSurf is modular software for global analysis of fluorescence spectroscopic data. The graphical user interface of Chisurf can be separated into three regions (a) the data reading / analysis dock, (b) a powerful interactive programming shell (an IPython prompt, https://ipython.org/) that accesses the currently running ChiSurf instance, and (c) the fitting region that gathers representations and plots of generated models / fits for diverse datasets, such as fluorescence correlation spectroscopy and fluorescence decay histograms (**Fig.1**).

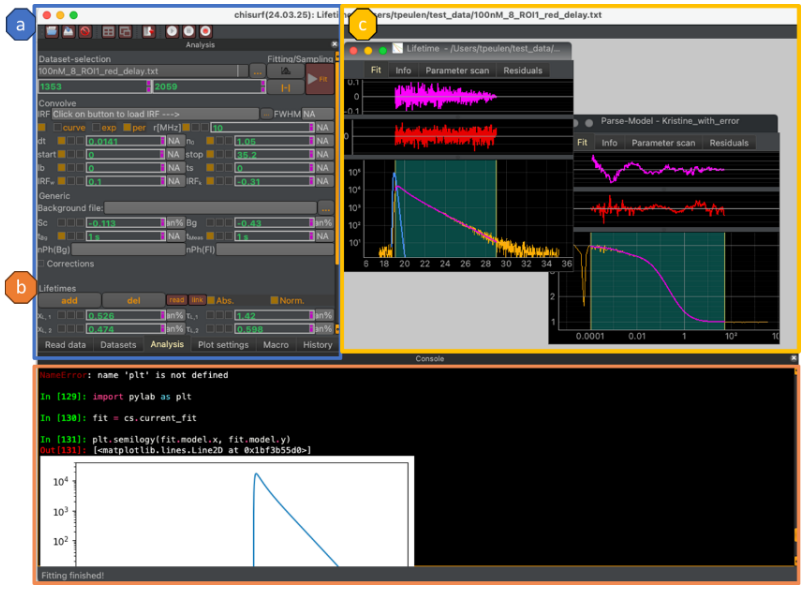


Fig.1 Graphical user interface of ChiSurf. The dock on the left (a) gathers user interfaces for reading data, creating data analysis, analyzing data, changing plots of created analysis, editing macros, and the history of the current program session. (c) An IPython programming shell interacts with the running ChiSurf session for interactive analysis, programming, and development. (c) Interactive graphical representations of the data facilitate an explorative data analysis that allows for introducing dependencies among datasets of different types for a joint analysis.

Integrated software modules for single-molecule spectroscopy, fluorescence correlation spectroscopy, and image spectroscopy (Fluorescence Lifetime Image Microscopy, FLIM) facilitate the joint analysis of imaging and single-molecule data, while the open Python programming interface allows for integrating other software for complex analysis (

Fig.2). The accompanying software can be used independently of the main analysis software.

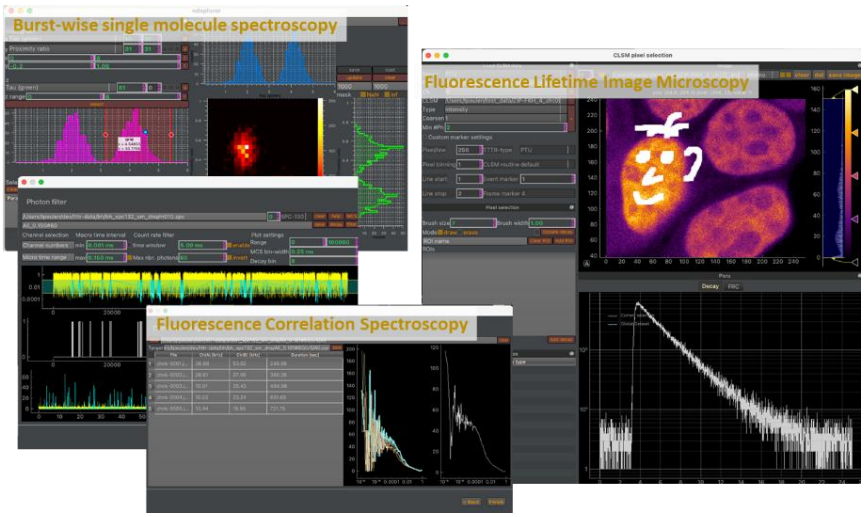


Fig.2 Accompanying software modules. ChiSurf is accompanied by software for burst-wise single molecule spectroscopy (ndXplorer), fluorescence correlation spectroscopy (tttrlib), and time-resolve image spectroscopy (clsmview), and additional software for statistical analysis. Historgams over multiparameter fluorescence spectroscopy data and sampled parameters can be computed for sub-ensemble analysis (Burst-wise single molecule spectroscopy). Photon traces can be analyzed for fluorescence correlation spectroscopy, FCS. Data collected on a microscope equipped with time-resolve detection can be processed for FLIM and for pixel-grouped analysis (Fluorescence Lifetime Image Microscopy, FLIM).

The main purpose of ChiSurf is the global analysis over multiple datasets (**Fig.3**). In ChiSurf this is achied by parameters of models (model parameters) for different data that can be "linked" for a joint/global data analysis. In global analysis multiple datasets are simultaneously described and dependencies of parameters across different datasets are exploited to maximize the accuracy and the precision of the analysis result, which can either be a point estimate determined by maximizing the agreement between the model and the data the variable parameters by fitting or by sampling over the variable parameters.

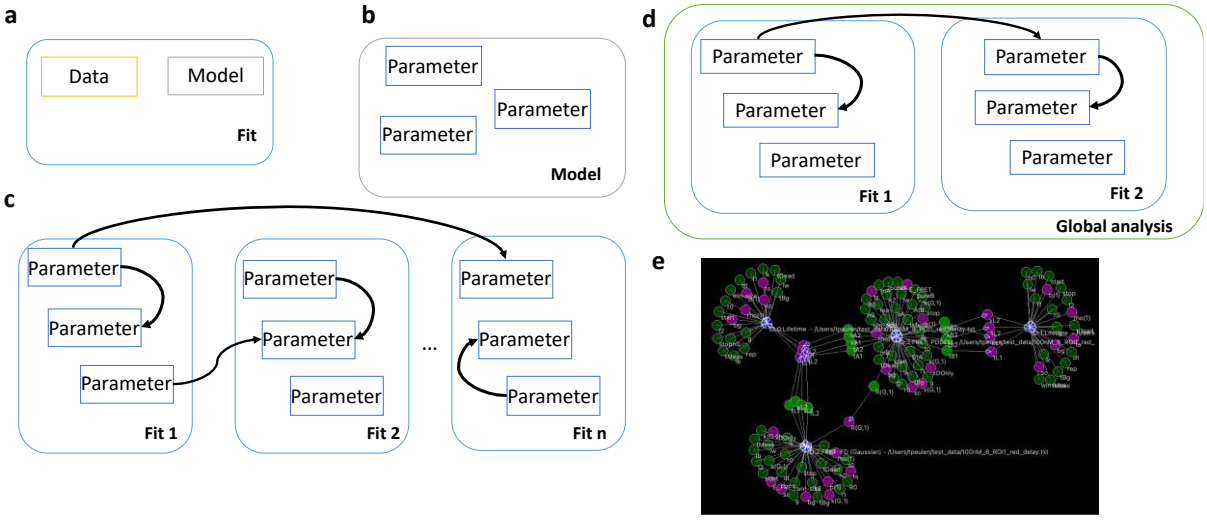


Fig.3 Global analysis in ChiSurf. (a) In ChiSurf data and models are combined to "Fits". (**b**) Each model has a set of parameters, which can be either a fixed parameter or a variable parameter. (**c**) The parameters of different "Fits" can be linked to introduce dependencies. (**d**) In a global analysis multiple fits with corresponding data and parameter dependencies are jointly analyzed, either by optimizing a scoring function or by sampling over the parameters. (**e**) Parameter dependency of a fluorescence decay analysis of the donor fluorescence decay in the presence and the absence of an acceptor, the direct excited acceptor, and the FRET sensitized acceptor. Parameter dependencies are visualized in a network graph.

This introduction gives a general overview and background information without providing specific information on how to use the software in particular use-cases for data analysis. The "Experiments" section provides specific information for experiments. The tutorial section at the end of this manual provides guides on how to use the software in particular use-cases. Code references are printed in a bold monospaced slab serif typeface, e.g., **`Example`**. This introduction provides background information and is best combined with a tutorial.

1.2. Data import

Internally, all imported data is managed in a single list (**`chisurf.imported_datasets`**). Elements in that list from the class **`chisurf.data.ExperimentalData`**. The user interface helps populating the dataset list (**Fig.4**).

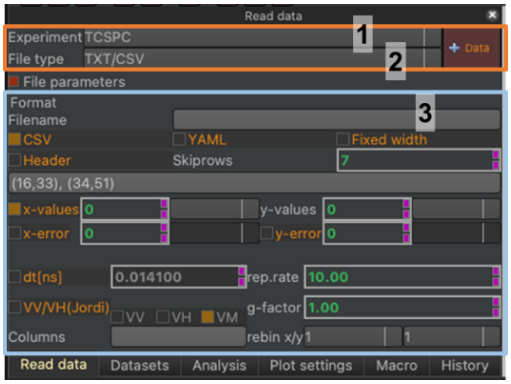


Fig.4 Data reading interface. The data reading interface consists of three regions. The dropdown menu in the first region defines the experiment type (1). The dropdown menu in the second region defines the file type for a particular experiment type (2). In the third region, the user can specify parameters for reading a particular file type (3). In the displayed example text/comma separated data of a time-correlated single photon counting (TCSPC) experiment is being read.

To read data using the graphic user interface, first select the corresponding experiment type (**Fig.4**, 1). Afterwards select the file type of the experiment (**Fig.4**, 2). Before reading data, check the parameters that are passed to the data read (**Fig.4**, 3). Finally, you can load the dataset into ChiSurf, either by clicking on the "**+Data**" button using the key combination "**Ctrl+N**" (Windows, Linux) or "⌘**+N**" on macOS. Alternatively, multiple files of the same kind can be opened in a single step by selecting the respective files in a file explorer of your choice and dragging the selected files into the user interface to the dataset list (**Fig.5**).

User and program actions correspond to actions in the IPython prompt.

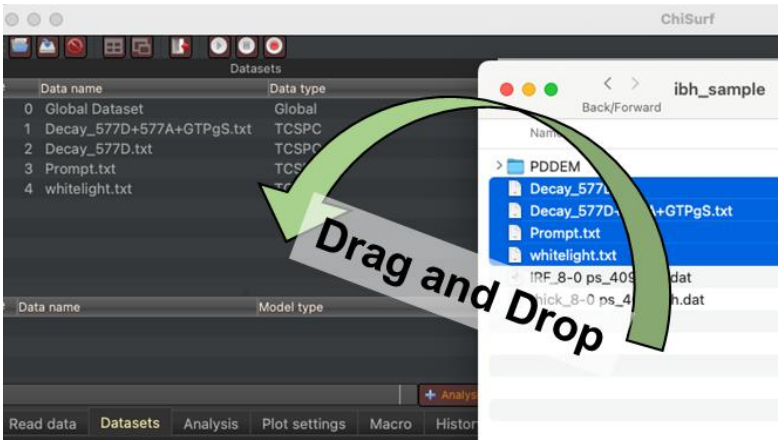


Fig.5 Drap and drop import of datasets. Files selected in a file explorer of your choice (here macOS Finder) can be opened by dragging selected files to the data set list.

Alternatively, files can be opened programmatically using the shell:

```
chisurf.macros.add_dataset(filename='/Users/tpeulen/dev/chisurf/test/data/tcspc/ibh_sample/Decay_577D+577A+GTPgS.txt')
```

The called macro will add open a dataset for the currently selected setup and with the current parameters for reading files. Programmatically, the current setup and the reading routing can be changed from the shell by assigning values to the **`cs.current_experiment`** and the **`cs.current_setup`** variable. For instance:

```
cs.current_experiment = 'FCS'
cs.current_setup = 'Seidel Kristine'
```

Interactions with the graphical user interface are reflecting as commands in the programming shell.

The context menu of the data list can be used to save, remove, group and ungroup datasets (**Fig.5**). Dataset of equal data type can be group into groups. Grouping dataset can be useful when analyzing datasets of similar type, e.g., when analyzing titration data.

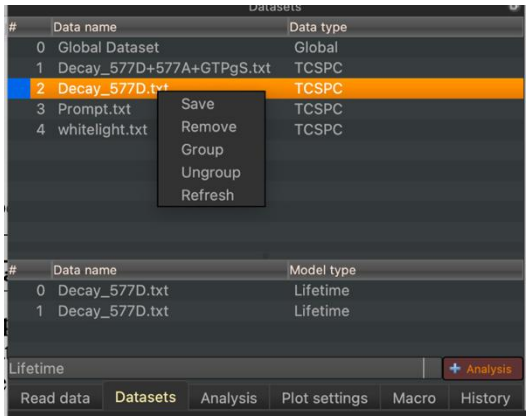


Fig.6 Dataset context menu. The context menu of the dataset list allows to save, remove, group, and ungroup datasets. The refresh option updates the data list.

To group datasets, select the corresponding dataset in the data set list and select in the context menu (right click) the group option (**Fig.7**).

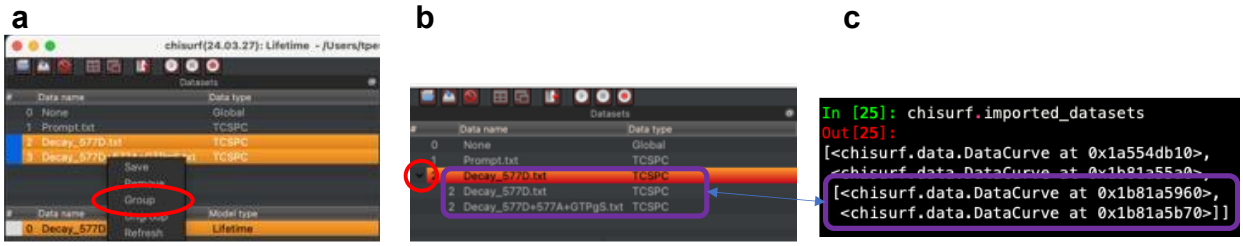


Fig.7 Grouping of datasets. (a) The context menu of the data set list allows to group datasets. (**b**). Grouped datasets appear in the dataset list as a single tree entry that can be inspected in detail by unfolding the group (red circle). (**c**) in the programming shell grouped datasets are iterable objects.

In the shell datasets are grouped as follows:

```
chisurf.macros.group_datasets([1, 2])
```

Here, the numbers refer to the index of the dataset in the **`chisurf.imported_datasets`** list. The order of datasets in the list can be changed by editing the number of the dataset in the first column of the data list (**Fig.4**). The currently selected dataset can be accessed in the shell:

```
cs.current_dataset
```

Datasets that are curves be accessed by their attributes.

```
import pylab as plt
cs.current_dataset
plt.plot(cs.current_dataset.x, cs.current_dataset.y)
```

For details look at the Application Programming Interfaces, API, of ChiSurf.

1.3. Fit models

In ChiSurf a model combines (variable) parameters and approaches to compute theoretical data (forward model). In ChiSurf instances of models are usually tight to the data in a "**`Fit`**". The currently active fit, the corresponding data, and the model can be accessed in the shell.

```
cs.current_fit
cs.current_fit.data
cs.current_fit.model
```

Usually, "**`Fit`**" instances, and model instances are created jointly in the graphical user interface.

1.4. Creating fits

In ChiSurf a **`Fit`** combines data with a model (**Fig.3**). Fits are stored internally in the list **`chisurf.fits`**. New fits are added to this list using the graphical user interface in three steps (**Fig.8**).

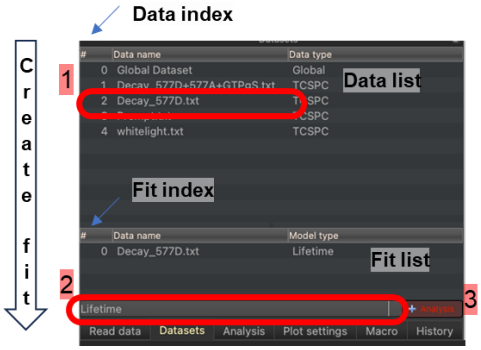


Fig.8 Creating fits / analysis in the graphical user interface. The top list displays imported data (Data list). The bottom list displays fits/analysis (Fit list).

In the first step, select a dataset in the data list by clicking on an item in the list (**Fig.8**, 1). In the second step, select a model for the selected dataset from the model selection dropdown menu (**Fig.8**, 2). In the third step, click on the '**+Analyis**' button next to the dropdown menu to create a new fit. In the first step, you can select multiple datasets of the same type by holding the Shift key while selecting data.

A new analysis can be created in the programming shell as follows:

```
chisurf.macros.add_fit(model_name='Lifetime ', dataset_indices=[2])
```

Here the number of the dataset can be a list of integers. The integers refer to the index in the **`chisurf.imported_datasets`** list. The order of fit can be changed by editing the number of the dataset in the first column of the fit/analysis list (**Fig.4**). Note, the order of the fits can matter, as parameters and variables of analysis are accessed through the fit index and the parameter name. Fits can also be created for grouped data sets. Selecting a fit in the fit list will active the corresponding fit windows.

1.5. Analysis dock

The analysis dock gathers variables and parameters of the currently active fit window (see **Fig.9**). The Analysis dock changes its content depending on the currently active fit (the activated fit window).

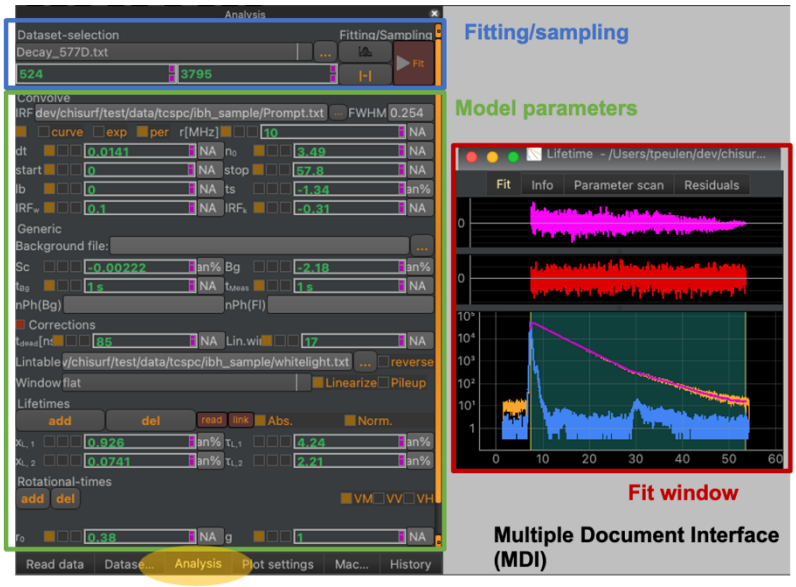


Fig.9 The "Analysis" dock. The analysis dock is activated by clicking on the yellow highlighted dock widget. The dock contains options to select different datasets, optimize/fit the free model parameters, adjust the fitting range (Fitting/sampling, **blue box**) in addition to the model parameters (**green box**) of the currently active fit, i.e., the active fit window (**red box**) in the MDI, Multiple Document Interface (gray background).

The analysis dock contains options for optimizing/fitting the currently active fit, adjusting the fitting range, and sampling over the free model parameter (see **Fig.9**). Moreover, the displayed dataset in a dataset group can be selected.

1.6. Parameters

A key attribute of parameters is their value. Parameter values are either fixed or variable floating-point numbers. Parameters can be part of a model. Variable parameters associated to a model instance are varied during model fitting/optimization and model sampling. A parameter can be bounded to restrict its range during sampling/optimization. An instance of a parameter can be connected (linked) to another parameter instance (**Fig.10**). A parameter that is linked to another parameter will report the value of the other parameter as its own value. This allows to introduce dependencies across different model instances.

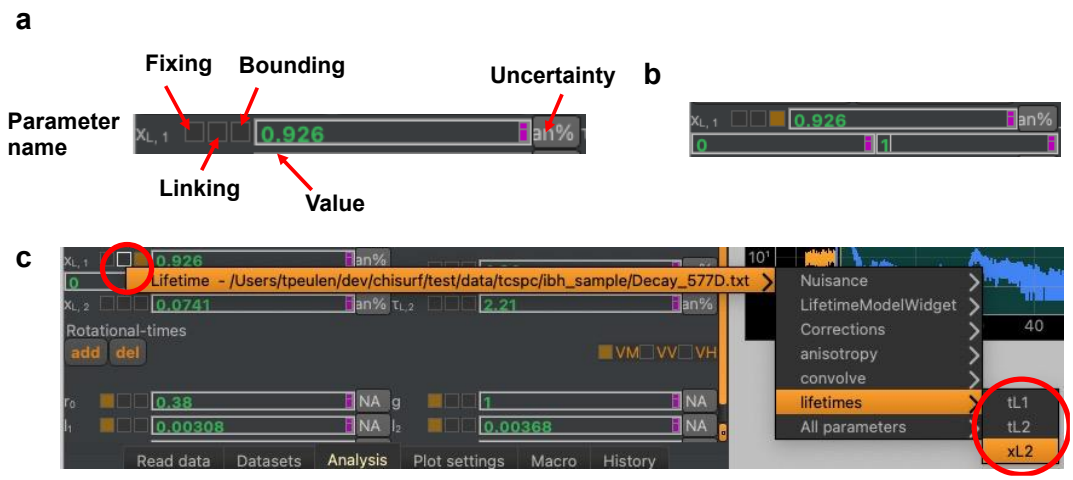


Fig.10 Fitting parameter. (a) Fitting parameters have a name (parameter name), can be free (variable parameter) or fixed (fixed parameter), can be linked to other parameters, bounded in region, and can have an optional uncertainty associated to them. The checkboxes next to the parameter name can be used to fix, link, bound parameters. (**b**) Bounded parameters display the lower and the upper value of the parameter below. (**c**) Parameters can be linked to other parameters using the context menu of the linking checkbox. The context menu (accessed by a right click) displays the created fits in the ChiSurf instance. The submenus of the context menu display parameter groups and parameters (red circle to the right).

Parameter values are by modifying the parameter value displayed in the user interface (**Fig.10**, **a**). Parameters are fixed using the first checkbox of the graphical parameter control interface (**Fig.10**, **a**). The second checkbox allows to link a parameter to another parameter. The second checkbox from the left and its tooltip report on the linking state of a parameter (**Fig.10**, **a**). The third checkbox from the left enables parameter bounds (**Fig.10**, **a, b**).

Actions in the user interface on parameters can be called from the shell. In the ChiSurf shell script below, two parameters are created, values are assigned to the respective parameters, and parameters are linked to each other, to illustrate how to create and interact with parameters.

```
p1 = chisurf.parameter.Parameter(name='p1', value=0)
p2 = chisurf.parameter.Parameter(name='p2', value=0)
p1.value = 1
p2.value = 2
p1.link = p2
p1.value == 2 # True
```

Links are removed by assigning `None` to a link attribute.

```
p1.link = None
p1.value == 1 # True
```

The variable parameters of the current fit model are accessed using the `parameter_dict` attribute.

```
cs.current_fit.model
cs.current_fit.model.parameter_dict
```

All parameters (fixed & variable) of the current fit model are accessed using the `parameters_all_dict` attribute.

```
cs.current_fit.model.parameters_all_dict
```

Parameter bounds can be enabled, assigned and disabled in the shell by setting the `bounds_on` attribute and assigning upper and lower bounds. Here, is an example for the parameter named '`sc`'.

```
chisurf.fits[0].model.parameters_all_dict['sc'].bounds_on = True
chisurf.fits[0].model.parameters_all_dict['sc'].bounds = (0.0, 1.0)
chisurf.fits[0].model.parameters_all_dict['sc'].bounds_on = False
```

Parameters can be fixed and made variable parameter as follows:

```
chisurf.fits[0].model.parameters_all_dict['sc'].fixed = True
chisurf.fits[0].model.parameters_all_dict['sc'].fixed = False
```

Fixed parameters are not varied during fitting and sampling.

1.7. Fit interface

The analysis dock displays an interface for sampling and optimization for the currently active window in the fit window Multiple Document Interface (MDI) (**Fig.8**). The fitting and sampling interface allows to select the current dataset in a data group, replace a dataset with another loaded dataset, sample over variable model parameters, fit/optimize variable model parameters, and adjust the model range (**Fig.11**).

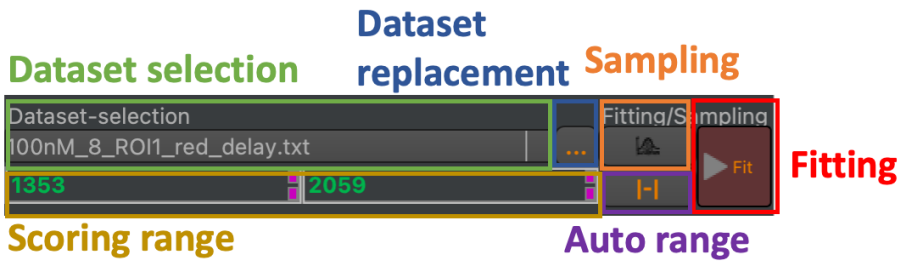


Fig.11 Elements in the dataset optimization and sampling interface. The interface can be used to select the currently displayed dataset (green box), replace the current dataset (blue box), sample over the variable model parameters (orange box), optimize/fit variable model parameters to a dataset group (red box), adjust the range of the scoring function (yellow box), and auto adjust the scoring range (purple box).

1.8. Model scores

The fitting minimizes the difference between the model and the data. Sampling samples over variable (free) model parameters to estimate probability distributions over parameters. Both, sampling, and optimization require a score. The score depends on the model, the data, and the data noise. The value of a score can of the current fit can be accessed as follows:

```
fit = cs.current_fit
fit.get_score(score_type='chi2')
```

Here, the `score_type` defines the type of the score. Default score types that must be implemented for all fits/model combinations is the sum of squared data-noise weighted deviations between the model and the data, '`chi2`', and `'chi2r'`, the sum of squared data-noise weighted deviations reduced by the degrees of freedom (the number of observations - number of model parameters). Values of '`chi2`' and `'chi2r'` can also be accessed as follows:

```
fit = cs.current_fit
fit.chi2
fit.chi2r
```

For a `fit` object the code

```
fit.get_wres()
```

returns the deviation between the data and the model weighted by the data noise.

1.9. Model scoring range

The value of the score depends on the data that is used for scoring. Often, the model is scored in a particular data range. In cases where the data are curves, the scoring range is defined by an upper and lower value (fit range). In the graphical user interface, the scoring range can be adjusted in the 'Data optimization & sampling interface' of in a plot of the data and the model (**Fig.12**).

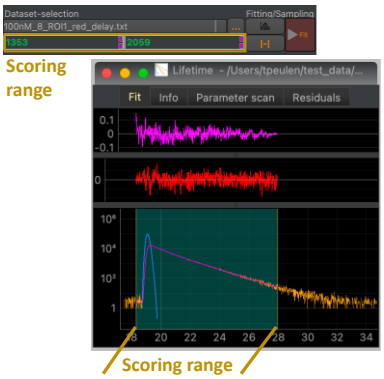


Fig.12 Adjusting the scoring range. The scoring range can be adjusted using inputs for the lower and upper bound of the scoring range in the data optimization and sampling interface (top). Alternatively, the scoring range can be adjusted in plots of the data and the model (bottom).

In the programming shell the fit range of the current fit is adjusted using integers as lower and upper bounds that correspond the index of the data.

```
fit = cs.current_fit
fit.fit_range = 61, 649
```

In this example, `61, 649` is the lower and upper bound, respectively.

1.10. Parameter optimization

ChiSurf uses the `lmdif` and the `lmder` algorithm implemented in `MINPACK` to optimize variable parameters. Variable parameters are optimized either by clicking on the 'Fit' button in the data optimization and sampling interface (**Fig.13**) or using the shell:

```
fit = cs.current_fit
fit.run()
```

Grouped fits offer the option to first optimize local variable parameters before optimizing the global fit.

```
fit = cs.current_fit
fit.run(local_first=True)
```

The effect of optimizing (fitting) variable model parameters to data for a fluorescence decay curve are displayed in **Fig.13**.

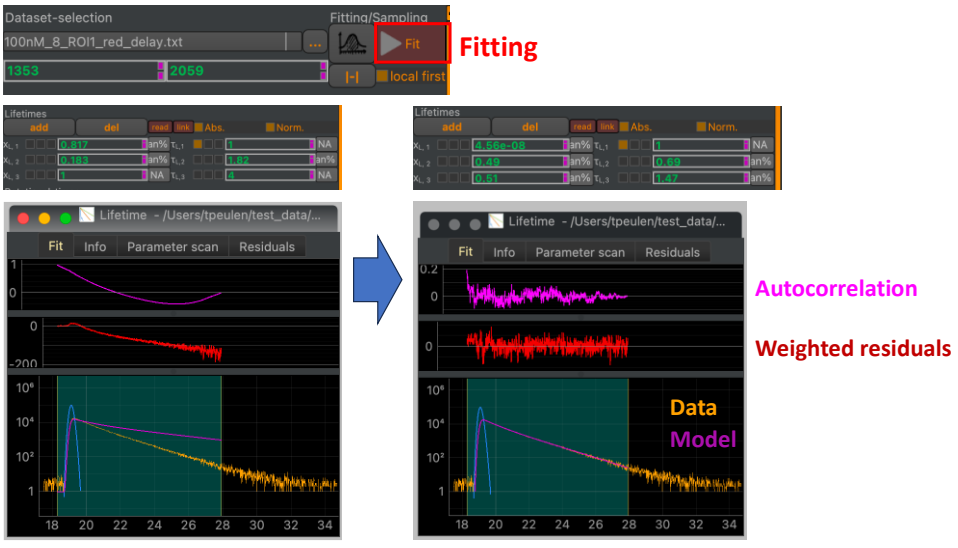


Fig.13 Optimizing variable parameters. The Fit button (red box) optimizes the agreement between the model and the data. The middle panels display fixed and variable model parameters before and after fitting (clicking the 'Fit' button). The bottom displays the data and the model before and after fitting. The autocorrelation of the weighted deviations between the data and the model weighted by the data noise (weighted residuals) and the weighted residuals visually captures the similarity between the data and the model.

```
Parameters of the optimization algorithms can be defined in the optimization
section of the ChiSurf settings file (optimization:
  global_threaded_model_update: false
  global_optimize_local_first: false
  leastsq:
    ftol: 1.49012e-08
    xtol: 1.49012e-08
    gtol: 0
    maxfev: 0
    epsfcn: 0
    factor: 100
    full_output: true
  mem:
    lower_bound: 1.0e-08
    upper_bound: 10000000
    maxiter: 150000
    maxfun: 1500000
    factr: 10
    reg_scale: 1
  sampling:
    method: emcee
    steps: 1000
    thin: 1
    chi2max: 1000000000
    n_runs: 10
```

Fig.14).

```
optimization:
  global_threaded_model_update: false
  global_optimize_local_first: false
  leastsq:
    ftol: 1.49012e-08
    xtol: 1.49012e-08
    gtol: 0
    maxfev: 0
    epsfcn: 0
    factor: 100
    full_output: true
  mem:
    lower_bound: 1.0e-08
    upper_bound: 10000000
    maxiter: 150000
    maxfun: 1500000
    factr: 10
    reg_scale: 1
  sampling:
    method: emcee
    steps: 1000
    thin: 1
    chi2max: 1000000000
    n_runs: 10
```

Fig.14 Optimization section in settings file. Optimization parameters for the optimization alogrithms are gathered in the optimization section of the ChiSurf settings file.

The ChiSurf settings file is described in more detail in **Section 4**.

1.11. Parameter sampling

ChiSurf uses emcee to efficiently sample over free model parameters. The sampling over the free model parameters can initiated using the "Distribution" button next to the "Fit" button in the fitting and optimization interface (**Fig.15**).

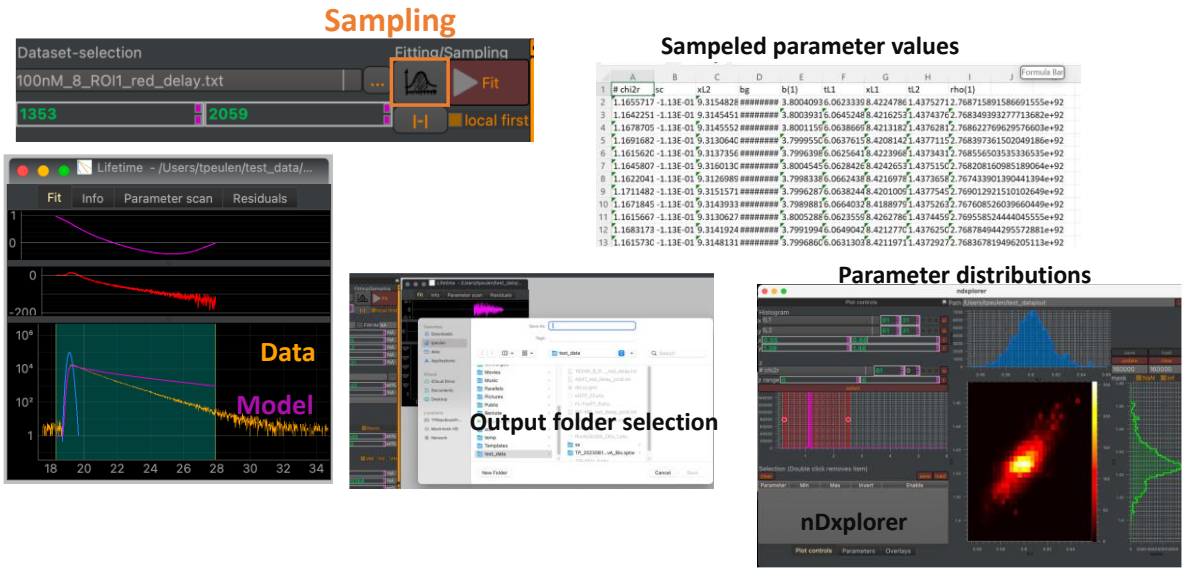


Fig.15 Sampling over variable model parameters illustrated for time-resolved fluorescence analysis. Data and model of a fit are displayed below. Clicking on the sampling button (highlighted in orange) initiates the sampling over the free model parameters and opens a folder selection menu (bottom). During the sampling ChiSurf will become inactive and create output files (txt files) containing the sampled model parameters. The outputted files can be opened in tools for multidimensional histograms such as nDxplorer (part of ChiSurf) or Margarita (Seidel software) to visualize distributions of the sampled parameters (bottom right).

The settings controlling the sampling, such as the number the number of steps, are located setup in the "sampling" section of the ChiSurf settings file (**Fig.14**).

The sampling can also be initiated from the shell. Variable parameters of the currently active fit are sampled as follows:

```
fit = cs.current_fit
chisurf.fitting.fit.sample_fit(fit, filename="/output_path/outfile.er4")
```

The settings of the sampling from the shell can be adjusted using the parameters of the `sample_fit` function (**Fig.16**).

```
def sample_fit(
        fit: Fit,
        filename: str,
        method: str = 'emcee',
        steps: int = 1000,
        thin: int = 1,
        chi2max: float = float("inf"),
        n_runs: int = 10,
        step_size: float = 0.1,
        temp: float = 1.0,
        **kwargs
)
```

Fig.16 Definition of the sample fit sample function. The function can be used for sampling over variable fit parameters.

Note, for an accurate analysis the sampling must be as complete as possible. For complex high dimensional models, the number of steps must be adjusted.

1.12. Multiple document interface

The multiple document interface displays fits created using the graphical user interface as windows. When new fits are created in the user interface, corresponding windows will in the MDI, Multi Document Interface (**Fig.17**). Fit documents in the MDI are windows that can be freely positioned, minimized, and closed. Closing a document closes the corresponding instance of a fit.

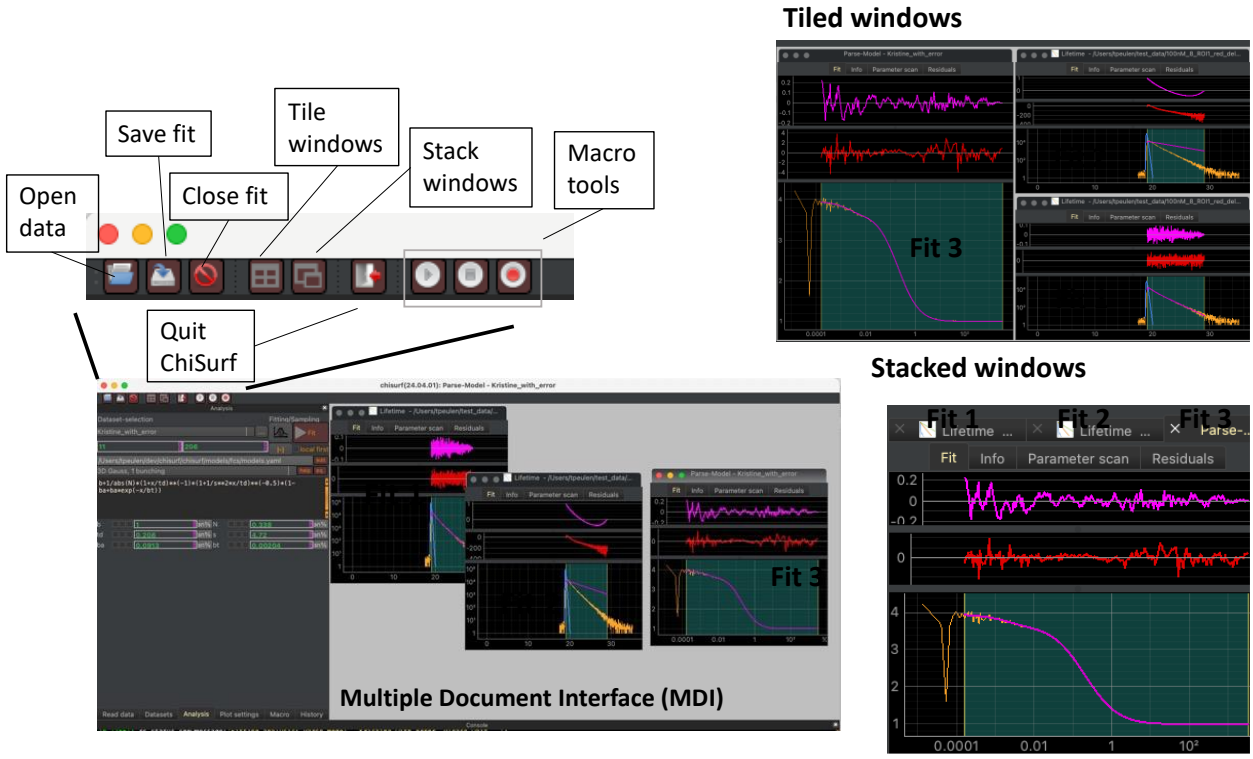


Fig.17 Elements controlling windows in the multi document interface. The toolbar in the of a ChiSurf window can be used to open data (with the current data reader), save fits to files, close the currently open fit, tile windows in the multi document interface

(MDI), stack windows in the MDI, and to control macros. When windows are tiled, all windows will be displayed in the MDI at once (top right). When windows are stacked a toolbar in the MDI controls which window is currently displayed in the MDI.

Moreover, fit documents can be tiled and stacked in the MDI using the functionality in toolbar (**Fig.17**). Activating selecting another fit document in the MDI calls:

```
cs.current_fit = chisurf.fits[0]
```

in the shell. In the example, the index "0" refers to the fit in that corresponds to the selected fit window.

1.13. Fit Plots

Fits display (depending on the model) different plots that can be selected within the fit window in the MDI (**Fig.18**). Plot settings are displayed in a separate dock window. The displayed settings depend on the currently selected plot.

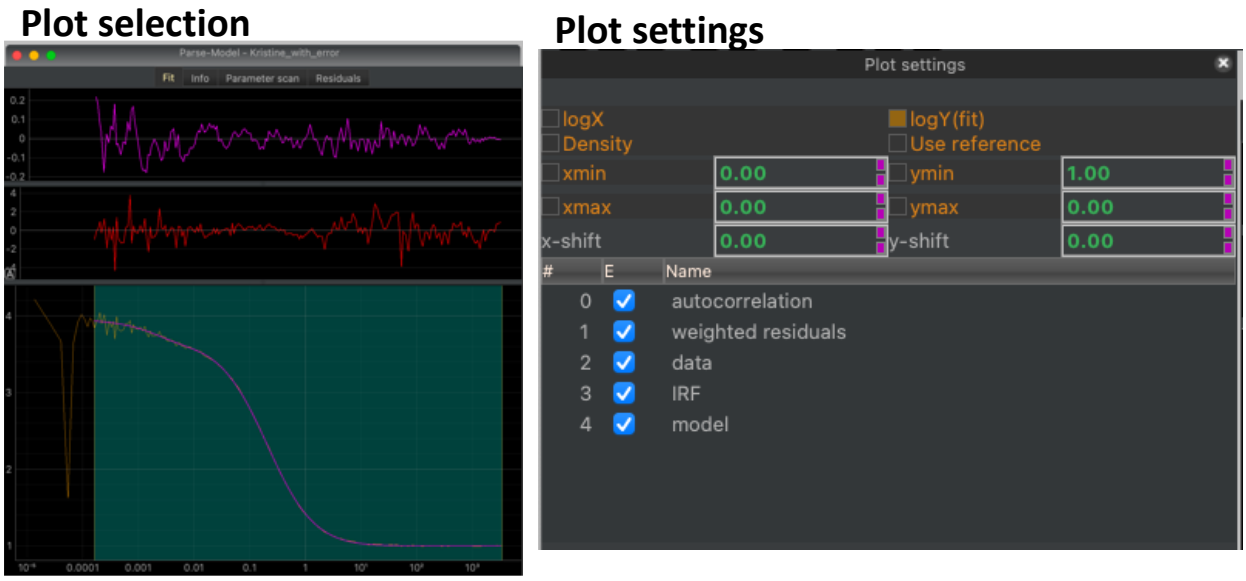


Fig.18 Plots in fit windows. The fit windows in the multiple document interface display different plots. The plots created for a certain model can be selected on the of a fit window (left, highlighted in yellow). The settings of the currently selected plot of the active fit window are displayed in the 'Plot settings' dock (right).

Most fit models display line plots of the corresponding data and model function along with weighted residuals and auto correlation functions computed of the weighted residuals (**Fig.18**, left). The settings of a plot can be used to modify the plot (**Fig.18**, right).

1.14. Parameter scan

A Chi Square support plane analysis around the minimum (the fitted solution) can be performed in in fits/model that offer a "Parameter scan". In the parameter scan a free model parameter can be selected from

a dropdown menu. The selected parameter is varied in a defined range. Other free model parameters are optimized.

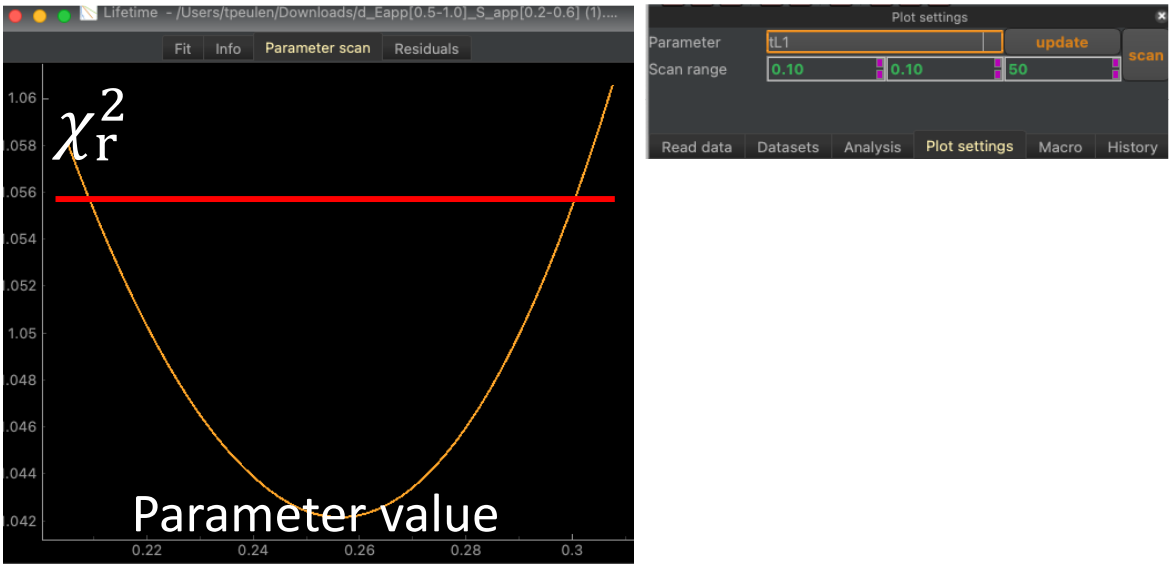


Fig.19 Parameter scan plots are offered by fits of certain models. The parameter scan varies a parameter in a certain range (top) and optimizes other free model parameters to create a χ_r^2 curve that depends on the parameter (bottom). The resulting χ_r^2 curve can be used to estimate uncertainties of model parameters. Red line chi2 confidence level 95% computed with F-Calculator.

This procedure (Support plane analysis) produces a χ_r^2 curve of the scanned parameter that can be used to estimate uncertainties (**Fig.19**). Upper limits of χ_r^2 can be computed using the F-Calculator tool (see Section 5.1, page 24).

Reduced Chi Square

$$\chi_r^2 = \frac{1}{n-p}\sum_{k=1}^{n}\left(\frac{I(t_k) - I_c(t_k)}{\sigma}\right)^2 \qquad (1)$$

2. Experiments

2.1. Time resolved fluorescence

Nuisances

Convolution. In time-resolved fluorescence experiments, usually, the model function is convolved with an instrument response function before comparing the model to the recorded data (iterative (re)convolution). Depending on the experimental conditions

and the used model function IRF convolution settings need to be adapted (

Fig.20). Certain convolution settings could be optimized during fitting or varied during sampling. However, usually, the convolution parameters are fixed. Nevertheless, convolution settings can be treated as a variable model parameter.

Fig.20 Convolution parameter group box. (1) Selection of experimental instrument response function, IRF. If no experimental IRF is selected the convolution uses synthetic (computed) IRF. (2) Full width half maximum of IRF. (3) Enable/disable IRF convolution with model function. (4) use direct convolution of fluorescence decay with IRF (use for parse models). (5) use fast exponential convolution (lifetime models), (6) use fast periodic exponential convolution (lifetime models). (7) repetition rate of laser in MHz (used for periodic convolution). (8) bin width of decay histogram. (9) scaling factor for fluorescence decay (number of photons) if fixed area of model function in fit region is scaled to experimental data (considering the data noise). (10) convolution start/stop. (11) background of instrument response function. (12) time-shift of IRF. (13) width of synthetic IRF. (14) shape-parameter (skewness) of synthetic IRF.

The primary challenge in time-resolved fluorescence experiments lies in the convolution with the instrument response function (IRF). Corrections are necessary to account for periodic excitation, and adjustments to parameters such as convolution start and stop points, as well as scaling of the model fluorescence decay to match the data (optional auto-scaling), are crucial. Incorporating background into the IRF and accounting for time shifts of the IRF further complicate the process. When an experimental IRF is unavailable, a synthetic IRF can be generated, often modeled as a skewed normal distribution. Parameters such as width and skewness of the synthetic IRF can then become free (variable) parameters during fitting and sampling, offering greater flexibility in the analysis process.

Background. The background is another significant nuisance in time-resolved fluorescence experiments. The fluorescence intensity is a combination of fluorescence signal and background components. Background can include constant elements, such as afterpulsing over long time scales, as well as other sources like scattered light, which is especially prominent in samples with weak fluorescence and strong

scattering. Effectively accounting for background is essential for accurate analysis and interpretation of fluorescence data.

Fig.21 Background and generic settings. (1) Selection of a background file, (2) scatter pre-factor, (3) constant background offset, (4) background acquisition time, (5) measurement acquisition time, (6) compute number of background and fluorescence photons

Various options exist for modeling the background: (1) incorporating scattering effects into the instrument response function, (2) including a constant offset to account for dark counts, and (3) employing a patterned offset (**Fig.21**). The acquisition times of both the background file and the experiment are essential, as they influence pile-up corrections. These acquisition times are utilized to compute the number of photons contributed by both the background and the fluorescence. The model used is typically a combination of the background and fluorescence components.

Additional corrections. TCSPC data can suffer from pile-up and differential non-linearities, which distort measurements and compromise accuracy. In this context, pile-up refers to the phenomenon where multiple photons arrive within the same time bin, while differential non-linearities, DNL, arise due to the system's response varying the time since the last sync pulse. Systems perturbed by DNLs show correlations for uncorrelated light. Considering these artifacts is crucial for extracting reliable information from TCSPC data.

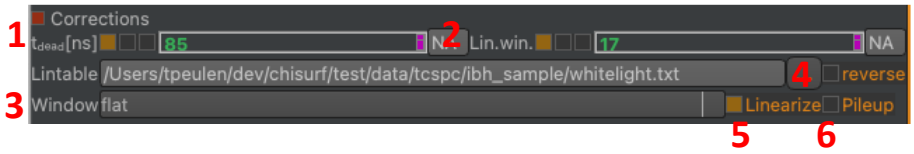


Fig.22 Additional corrections. Parameters to correct for pile-up and instrumental differential non-linearities, DNL (1) instrument dead-time, (2) window size for smoothing experimental linearization table, (3) window function for computing smoothed linearization table, (4) option to reverse linearization table, (5) option to enable/disable DNL correction, (6) option to enable/disable pile-up correction.

ChiSurf offers options to consider pile-up and DNLs. Instead of modifying the acquired data, the model function is perturbed to preserve the counting statistics for accurate error estimates.

Equation parsing

ChiSurf Equation Parser enables the computation of fluorescence decay according to any specified function, with equation parameters serving as model parameters. Users can access a list of predefined equations, which are editable to suit their needs (**Fig.23**).

Fig.23 Fluorescence decay equation parsing. (1) A JSON file with a set of pre-defined equations for time-resolved fluorescence is shipped with ChiSurf. (2) The edit button opens a (3) JSON file editor that can be used to edit the pre-defined equations. (4) A dropdown menu lists the pre-defined equations. (5) Descriptions/help on the pre-defined equations can be enabled/disabled. (6) Parse button to update the model when equation was edited. (7) Button to enables/disables displaying the an editing equations. (8) equation editor. (9) Representation of parameters extracted from equation.

The Equation Parser operates by generating a curve for the equation on the time-axis, with the variable 'x' representing time. This curve is then convolved with the Instrument Response Function (IRF). ChiSurf offers various convolution modes, including fast convolution, fast periodic convolution, and curve convolution. The latter mode is slower but necessary when parsing equations. Additionally, ChiSurf handles other nuisances in a similar manner as with fluorescence lifetime-based model functions.

Fluorescence Lifetime

In fluorescence lifetime models, the most widespread type of model involves the linear combination of exponential decays. This approach is used in various applications such as anisotropy, FRET, fluorescence lifetimes, distance distributions, and more, where they are all modeled as combinations of exponential functions.

$$f(t) = \sum_{i=1}^{n} a_i \exp\left(-t/\tau_i\right) \tag{2}$$

A special form of this model provides additional options for convolution with the Instrument Response Function (IRF). These options include "fast convolution," "fast periodic convolution," and the convolution of the IRF curve with the model. Furthermore, additional AVX-optimized convolutions are available for enhanced computational efficiency.

Lifetime mixtures

Lifetime mixture models combine fluorescence lifetimes "spectra," which consist of pairs of amplitudes and lifetimes from different models. These models are combined using a weighted sum approach, where each model's contribution is scaled by a corresponding weight. This allows for the representation of complex fluorescence behaviors by blending the characteristics of multiple underlying models.

FRET Models

Discrete FRET rate constants

In the ChiSurf discrete FRET rate constant model, the fluorescence decay is described by discrete fluorescence lifetimes and discrete FRET rate constants. This model accounts for the heterogeneity in energy transfer processes by allowing each donor-acceptor pair to have its own specific FRET rate constant, reflecting the varying efficiency of energy transfer among different donor-acceptor pairs. ChiSurf uses a homogeneous FRET model that assumes that all fluorescence lifetimes are quenched by the same FRET rate constant. This simplification treats the system as if all donor-acceptor pairs exhibit identical energy transfer dynamics, resulting in a uniform quenching of fluorescence lifetimes across the entire population of donor molecules.

FRET induced donor decay. Linear combination of exponential decays where characteristic times are rate constants of the energy transfer and amplitudes are population fractions.

$$\epsilon_D(t) = \sum_{j=1}^{n} c_j \exp\left(-t \cdot k_{RET,j}\right) \tag{3}$$

$$f_D^{(D0)}(t) = \sum_{i=1}^{n} a_i \exp\left(-\frac{t}{\tau_i}\right) \tag{4}$$

$$f_D^{(DA)}(t) = f_D^{(D0)}(t) \cdot \epsilon_D(t) \tag{5}$$

$$k_{RET,j} = 1/\tau_0 \left(\frac{R_0}{R_{DA},j}\right)^6$$

Mixture of normal distributions

$$\epsilon_D(t) = \sum_{i=1}^{n}\sum_{j=1}^{m} c_{\blacksquare} \exp\left(-t \cdot k_{RET}(R_{DA})\right) \quad (6)$$

Partial donor-donor energy migration

Missing

Worm-like chain

The worm-like chain model is a simplified representation of polymer chains, including proteins, in which the chain is treated as a series of interconnected segments resembling a worm's body. This model is often used in the context of protein folding to describe the behavior of polypeptide chains as they fold into their native three-dimensional structures.

The worm-like chain model is a useful tool for conceptualizing the folding behavior of proteins and gaining insights into the fundamental principles governing the folding process. However, it's important to note that proteins are highly complex molecules, and while the worm-like chain model provides a simplified framework, it may not capture all the intricacies of protein folding in detail.

Reference curves

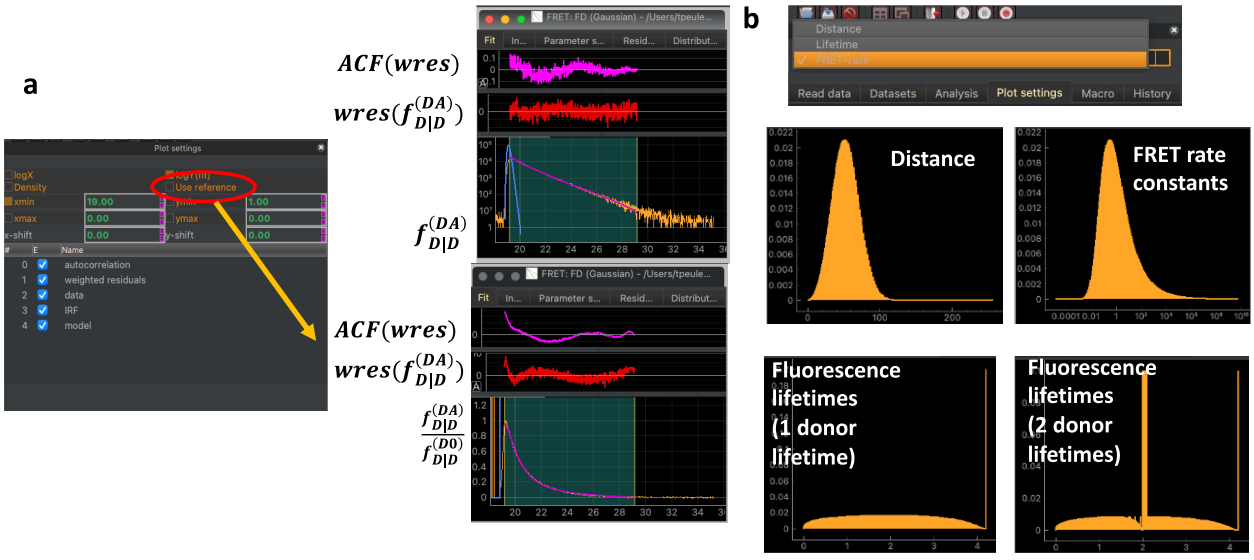


Fig.24 FRET model fit plots. (a) Fluorescence decays of the donor in the presence of an acceptor, $f_{D|D}^{(DA)}$, can be plotted using the fluorescence decay of the donor in the absence of an acceptor, $f_{D|D}^{(D0)}$, as a reference to compute the FRET induced donor decay, ϵ_D.

Enabling the "Use reference" checkbox displays the ratio $\epsilon_D = f_{D|D}^{(DA)} / f_{D|D}^{(D0)}$. (b) FRET models can be displayed as distance distribution, FRET rate constant distribution of as fluorescence lifetime distribution. Fluorescence lifetime distributions can adopt more complex shapes for multi exponential donor fluorophores.

The "Use reference curve" option in the plot settings of a FRET model (see **Fig.18** & **Fig.24**).

2.2. Fluorescence correlation spectroscopy

2.3. Global models

3. Scripts

3.1. FRET lines

4. Setting files

5. Tools

5.1. F-Calculator

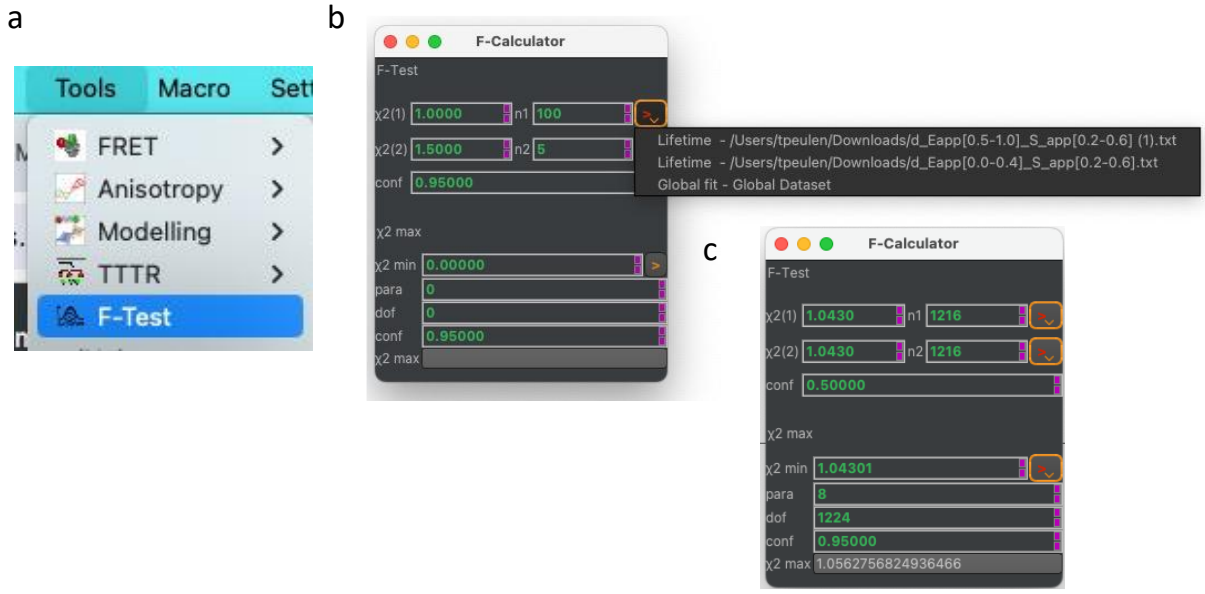


Fig.25 F-Calculator. Compute F-values for fits.

5.2. CLSMView

5.3. K2distribution

5.4. F-Test calculator

6. Plugins & Tools

6.1. nDxplorer

6.2. CLSMview

6.3. Anisotropy Wizard

The Anisotropy Wizard in ChiSurf is a guided tool designed to facilitate the setup and analysis of time-resolved fluorescence anisotropy data. It ensures that relevant parameters are correctly configured and linked for accurate analysis. Before importing data, configure the reading parameters to ensure correct interpretation of the input files. Verify data reading settings before importing files to avoid misinterpretation of the dataset.

This wizard helps in setting up analysis by linking relevant anisotropy parameters, correcting for sensitivity differences, and ensuring proper handling of mixing effects. The g-factor (g) is an essential correction parameter that accounts for differences in sensitivity between the VV and VH detection channels. Another crucial aspect is the mixing of anisotropy when using a high numerical aperture (NA) objective, which can introduce distortions in the measurement. Proper calibration and correction are necessary to ensure accurate anisotropy calculations. A detailed tutorial on how mixing factors and g-factors can be determined by reference measurements can be found in the Tutorial section of this manual.

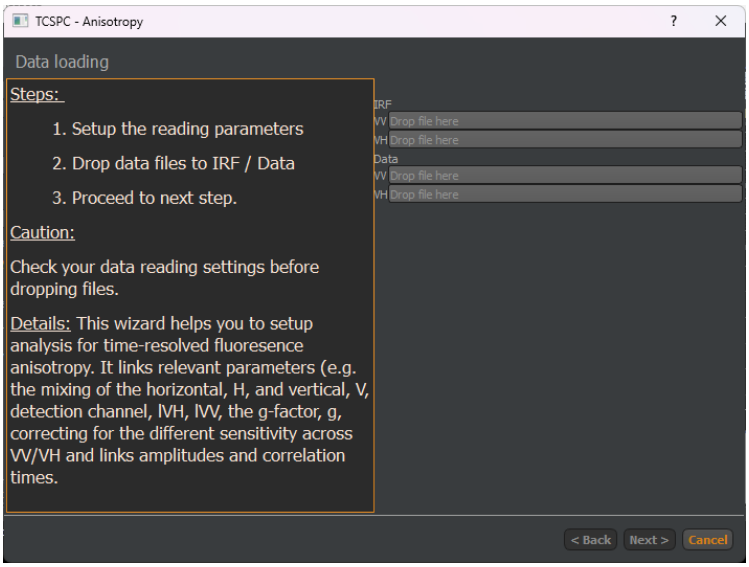


Fig.26 Data loading screen of Anisotropy Wizard. The text field on the left shows information on the current step of the wizard. The lines on the right accept files via drag-and-drop.

The first step in the analysis is reading data, which includes IRF and measurement data. The instrument

response function (IRF) must be independently provided for VV and VH channels. Both IRFs are used to correct time-resolved signals and extract precise anisotropy parameters. The files are provided via drag-and-drop. Note, make sure, that the settings of the reading routine are adjusted before dropping files into the wizard.

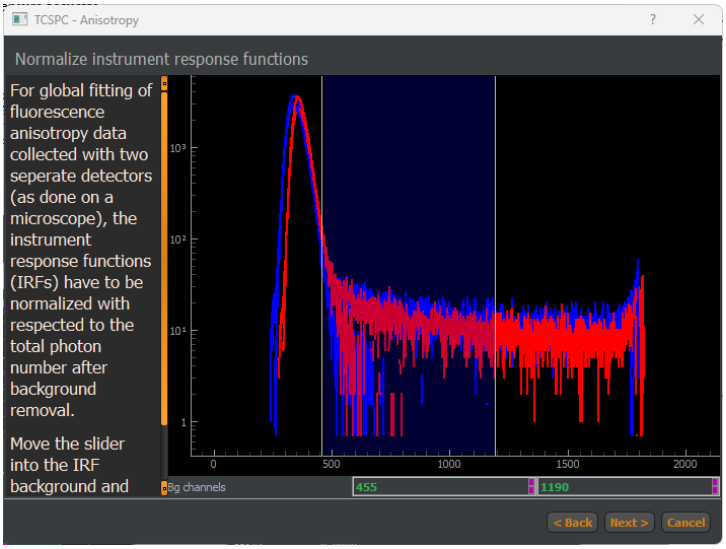


Fig.27 Instrument response function background correction. The slider can be used to adjust a background range for the instrument response function, IRF. The plot displays the IRF in parallel and perpendicular before and after background correction.

After reading the files, the instrument response function, IRF, is prepared for convolution, i.e., constant background is subtracted from the recorded IRF and the recorded IRFs in parallel (VV) and perpendicular (VH) are normalized.

Next, default values for the G-factor and the anisotropy mixing are read from the user folder and displayed in the next page of the Anisotropy wizard.

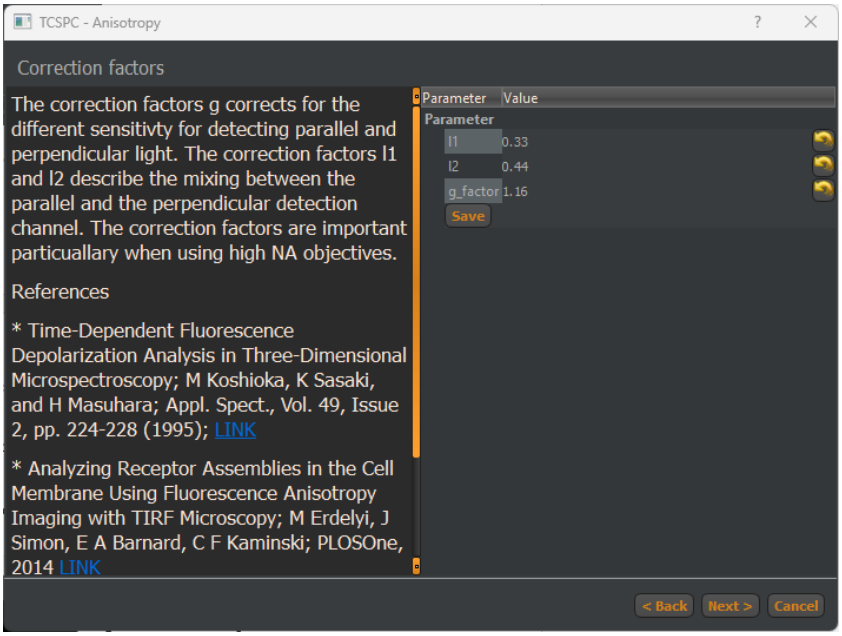


Fig.28 Adjustment of mixing parameters and g-factor. The parameters are read from the user folder and can be saved using the save button.

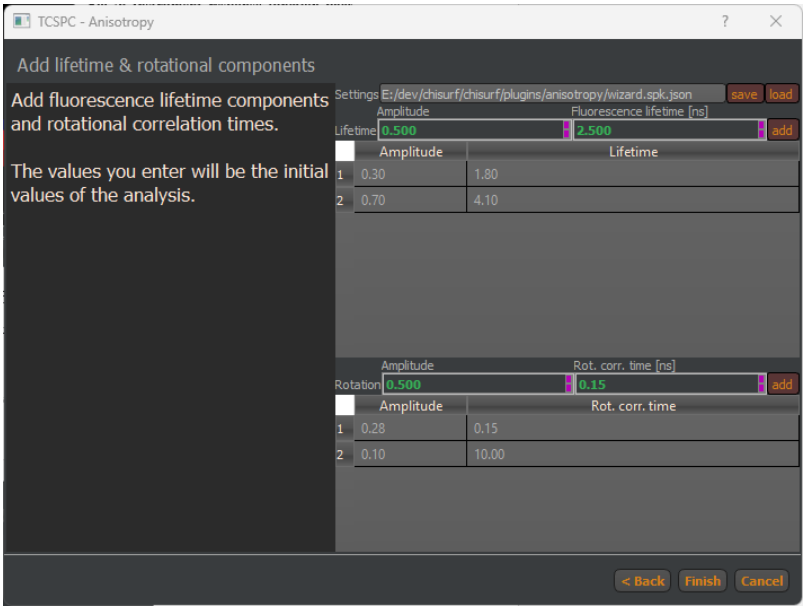


Fig.29 Definition of fluorescence lifetimes and rotational correlation times.

After defining the lifetime and anisotropy spectrum the "Finish" button closes the Wizard and creates a Global, a VV, and a VH fit window.

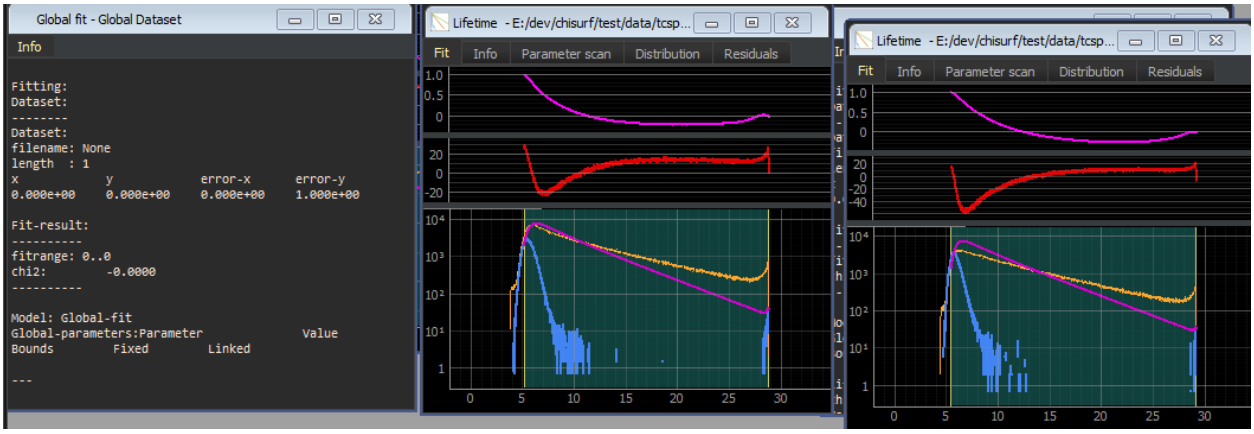


Fig.30 Created fit windows: Global, VV, and VH fit window

6.4. Correlation

Overview

The Correlator Plugin of ChiSurf provides an interactive user interface to functionality implemented in tttrlib (https://github.com/fluorescence-tools/tttrlib) to process single photon counting data and compute correlation curves following a 6 step workflow outlined in **Fig.31.** The correlator plugin is opened from the ChiSurf plugin menu (Plugins → Correlator).

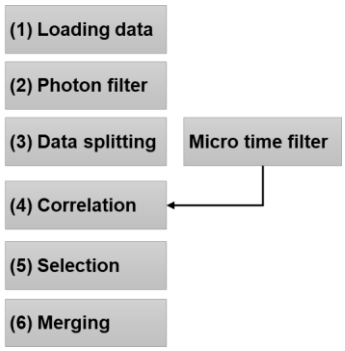


Fig.31 Workflow to compute fluorescence correlation spectroscopy curves. The loaded single photon counting, (1) **Data loading**, is filtered, **(2) Photon filter**, to select certain regions of the photon stream. For estimating uncertainties, the photon stream is separated into subsets, **(3) Data splitting**, and subsets of are correlated, **(4) Correlation**. To minimize artifacts correlation curves are visualized and selected, **(5) Selection**. Finally, a correlation curve is with associated uncertainties is computed, **(6) Merging**.

The workflow can be used to compute simple correlations and allows for more advanced filtering methods to enhance the contrast in fluorescence cross correlation spectroscopy, FCCS, and minimize artifacts. In the first step, the raw photon stream is opened. In the second step, filters are applied to the photon stream to mask photons that do not fulfil certain conditions, e.g., photons in regions of the stream where the count rate exceeds a certain threshold. In the third step, the selected / filtered photon stream in split into subsets. In the next fourth step, the subsets are individually correlated (optionally with a micro time filtered, e.g., for lifetime filtered correlation). Following the correlation, correlation curves are inspected and selected in the fifth step. To be finally merged into a joint correlation curve (**Fig.31**, Step 6, Merging).

Loading & plotting data

First, the photon stream data needs to be loaded. After opening a new Correlator from the Plugin menu, a new window is presented that allows you to open you photon stream (**Fig.32**).

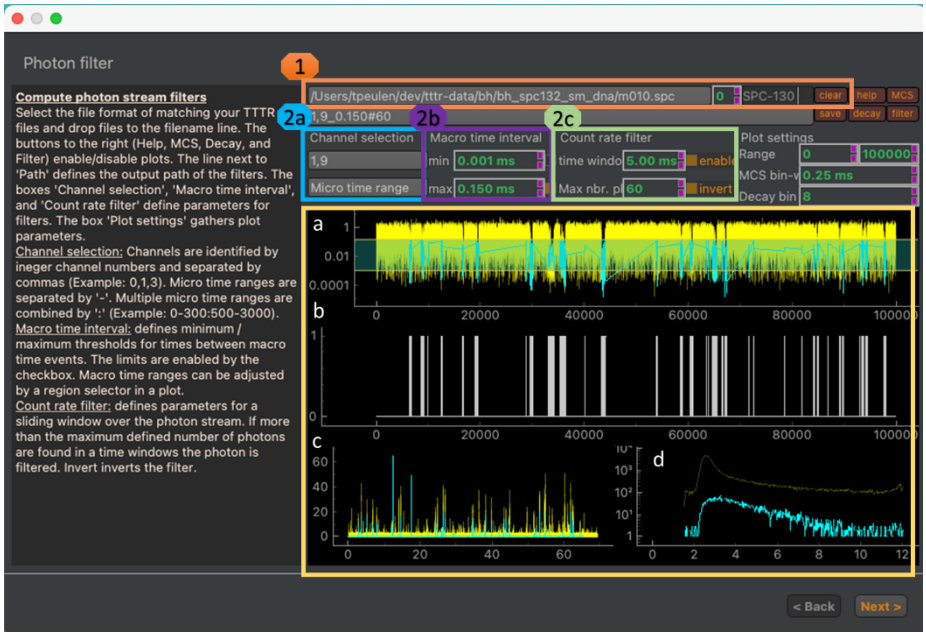


Fig.32. Data loading and photon filtering. Line widget, spin box, dropdown menu and clear button to load, select displayed data, select data reading routine, and clearing data, respectively (**1, orange box**). The help button to the right opens a text box displaying additional information. The MCS, decay, and the filter button enable and disable plots of the intensity trace, the micro time histogram, and the photon mask / filter, respectively. The bottom of the window (yellow box) displays plots of (**a**) the inter photon time of the raw photon stream (yellow) the selected photons (cyan), (**b**) the mask used to select the photons, (**c**) a histogram of the inter-photon time against the time of detection (intensity trace) of all photon in the current file (yellow) and the selected photon (cyan), and a (**d**) micro time histogram over all photons in the current file and the selected photons in a file. Channels and micro time range selections are entered as lists (**2a, cyan**). Filters on the inter macro time difference can be enable by checkboxes in the "Macro time interval" group (**2b, purple**). The filter values correspond to the region selector in section **a** of the macro time difference plot. Count rate related filter values are adjusted in the green box (**2c**).

To open files, first, select the reading routing. Next, select files in you preferred file browser (e.g. Windows Explorer, macOS Finder) and drop the selected filed to the text line to the top of the correlator window. After dropping the files to the text line, the files will be loaded into the correlator and the plots in the window will be filled. The loaded data can be cleared by clicking on the "clear" button (**Fig.32**). The photon selection mask can be saved using the "save" button on the top of the window.

Photon filter

The photon filter in Step 2 allows for selecting photon from the photon stream. The selected photons that are correlated in a later step. Photons can be selected by the detection channel, micro time ranges, and by applying count rate filters to the photon stream. Macro time differences between photons, the selection mask, intensity time-traces, and micro time histograms of all photons and selected photons are displayed in the photon filtering window (**Fig.32**).

Channel & micro time selection. The channel selection widget (**Fig.32, 2a**) is used to define selections based on the detector number of registered photons and to define micro time ranges. By default, all photons in the photon stream are selected. Specifying channel numbers restricts the selected photons to the specified channels. Details and examples on the channel and micro time selection are summarized in **Tab.XX1**.

Macro time interval. The time between two consecutively registered photons can be used as a filter to select high count rate regions in a photon stream. Thresholds on minimum and maximum time between two

consecutive photons are applied in the Macro time interval group (**Fig.32, 2b**). The selector in the macro time interval plot corresponds to the values set by the user in the "Macro time interval" group.

Count rate filter. For every photon in the stream photons within a define time window are selected. If for the selected photon and the defined time window more photons than are certain threshold (maximum number of photons) are found, the photon is not selected. The selection can be inverted to select high count rate regions.

Tab.1. Photon filter and plotting parameters.

Photon filter	Filter group	Example values	Description
Channel selection	Channel selection	0,8	Selects exclusively photon registered in channel 0 or 8.
Micro time selection	Channel selection	0-1000:1800-2200	Selects photons with micro times either in the range of 0 to 1000 or 1800-2200.
Minimal macro time interval	Macro time interval	0.001 ms	Minimum time between two consecutive photons
Maximal macro time interval	Macro time interval	0.128 ms	Maximum time between to consecutive photons
Time window	Count rate filter	5.00 ms	Time window to compute the count rate around a photon
Max nbr. Photons	Count rate filter	60	The maximum number of photons allowed in a time window
Invert	Count rate filter	Enable/Disable	If inverted photons in high count rate regions are selected. If disabled low count rate regions are selected.
Range	Plot setting	0, 10000	Defined plot range of the inter photon time / photon mask
MCS bin width	Plot setting	0.25 ms	Bin width used in the computation of the intensity trace/
Decay bin	Plot setting		Factor to coarsen the micro time before computing a histogram

The photon filter creates a folder for the intermediate steps of the analysis pipeline, e.g., the generated 'sl5' folder contains compressed JSON file with the filter parameters and the photon selection mask.

Correlator

The correlator uses the generated photon selection masks and the photon stream to compute correlation functions (**Fig.33**).

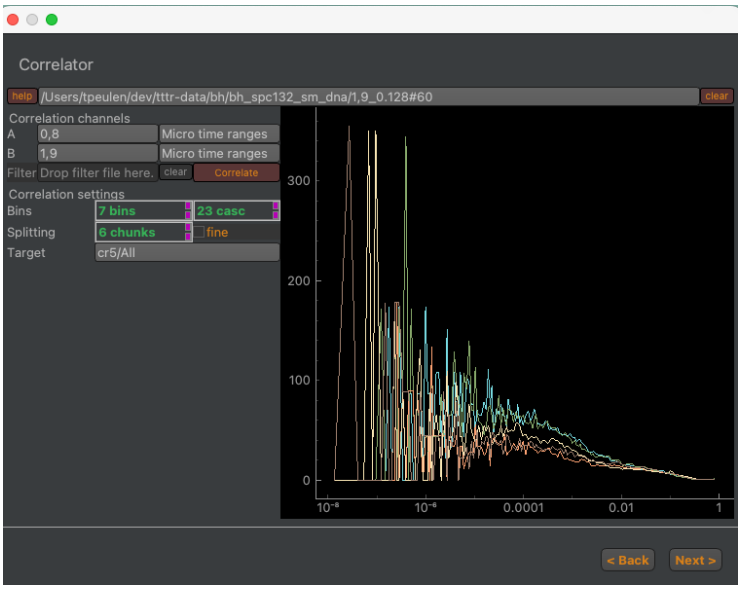


Fig.33 Interface of the correlation step in the correlation computation pipeline. Correlation channels and selected micro time ranges for the correlation are defined in the "Correlation" channel group. The number of correlation bins and the splitting of the photon traces into subsets are defined in the "Correlation settings" group. The number of bins per "cascade" defines the number of correlation bins. The "fine" checkbox enables a full correlation that utilizes the macro and the micro time in the correlation computation. The "Correlate" button initiates the correlation computations. Computed correlations are displayed in the plot to the right. The computed correlations are stored in the output folder defined in the "Target" field.

The correlation step computed for the photons selected by the photon mask correlation functions. The selected photons are split into subsets. For each subset a correlation curve is computed. The output of the correlation computation is stored for each subset in JSON files that refer to the photon filter and contain information on the correlation setting along with the corresponding correlation curves.

Correlation merging

Finally, the computed correlation curves can be merged (**Fig.34**). Before merging the correlation curves of the subsets into a joint correlation curve outliners (e.g. caused by aggregates) can be removed.

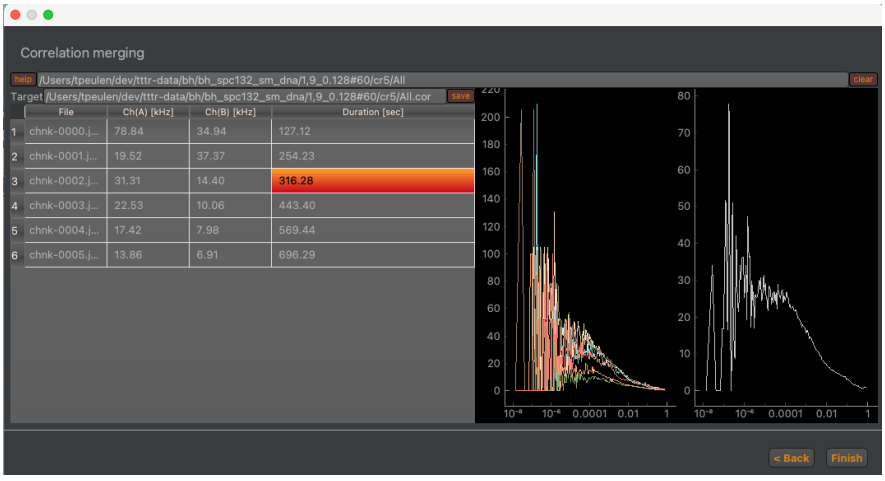


Fig.34. Selecting and merging correlation curves. The left side displays for every subset of the photon stream the count rate in the two correlation channels and the duration of the subset. Selecting a row in the left table selects the corresponding correlation curve displayed on the right-hand side. The merged correlation curve is displayed on the very right-hand side.

The curves of the subsets can be selected in the table on the left-hand side. A double click on the selected curve removes the curve from table and the merged correlation curve. The "Finish" button finalizes the pipeline. By default, the output of the pipeline is saved in the "cr5" folder.

6.5. Global view

Overview

In ChiSurf dependencies between parameters can be introduced by linking and visualized in graphs (**Fig.35**). The *Global view* plugin (*i*) visualizes parameter dependencies in directed graphs, (*ii*) saves parameter dependencies, and (*iii*) restores dependencies from files.

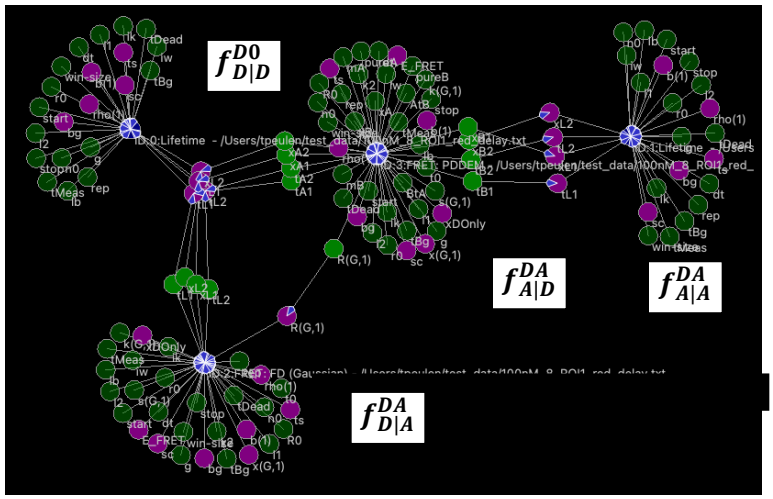


Fig.35. Parameter dependency graph in time-resolved fluorescence decay analysis. Fluorescence decays, f, that describe the donor fluorescence in the absence of an acceptor in a donor only, $f_{D|D}^{(D0)}$, the donor fluorescence the presence of an acceptor, $f_{D|D}^{DA}$, the acceptor fluorescence in a FRET sample $f_{A|A}^{DA}$, and the FRET sensitized acceptor fluorescence, $f_{A|D}^{DA}$. Parameters and models are represented by circles. Dependencies are illustrated by arrows. Parameters dependent on other parameters are colored in green. Fixed parameters are displayed in light green. Variable parameters are highlighted in magenta.

The *Global view* plugin opens from the Plugin menu, Plugins → Global view, in a separate window (**Fig.36**).

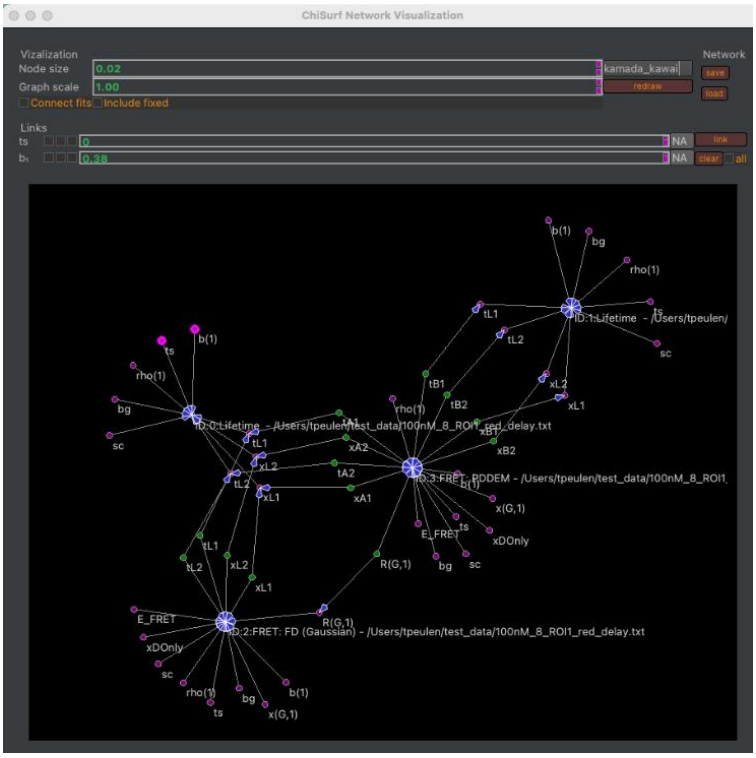


Fig.36. User interface of the *Global view* plugin. The plugin represents models and parameters in graphs (bottom). The Visualization group box of the plugin gathers options controlling the graph visualization (Node size, Graph scale). The Network group box can be used to save and load dependencies. The Link group box can be used to introduce and delete (clear) dependencies across selected and all parameters.

Changing visualizations

The visualization of the graph is controlled by widgets in the visualization box (**Fig.37**).

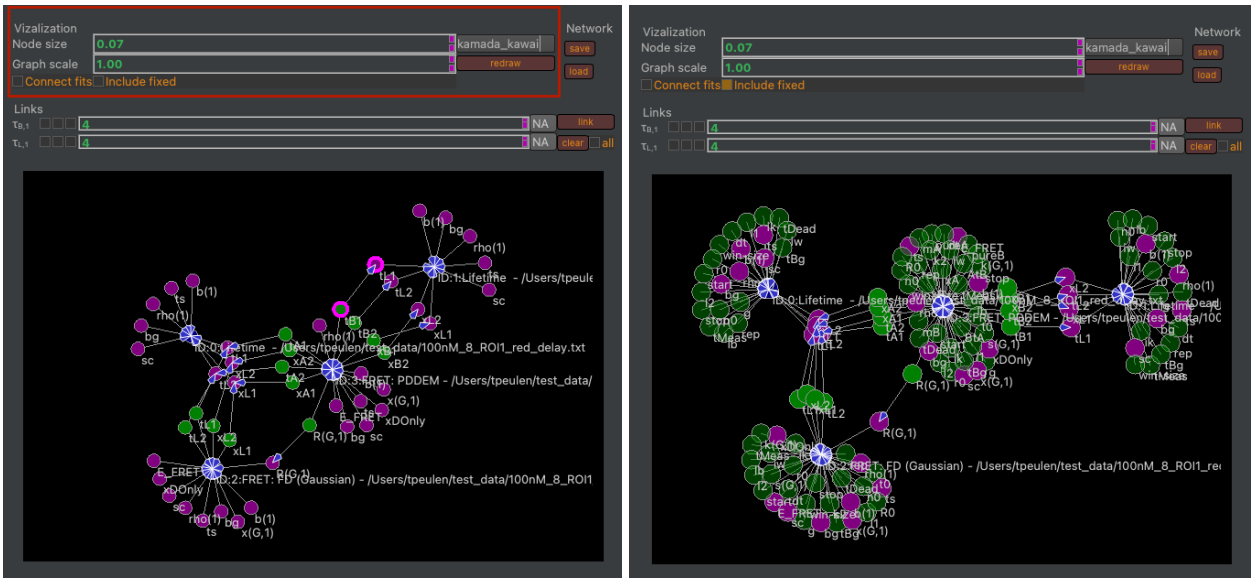


Fig.37. The graph visualization is controlled by the group box highlighted in red (left). The node sizes and the graph scale are controlled by respective spin boxes. The connect fits checkbox introduces additional edges between fits. The include fixed checkbox controls wether or not fixed parameters are displayed in the graph. The dropdown menu can be used to select a different algorithm for node placement.

The position of nodes can be controlled by dragging nodes of the graph.

Saving and loading networks

By default, the Global view plugin displays all models / fits in the current ChiSurf instance. Parameter networks can be saved using the save button (**Fig.38**).

Fig.38. Saving parameter dependency networks (left). Parameter dependency networks are saved in GraphML files (right).

ChiSurf stores fits in an ordered list. Hence, when loading networks file the fits in the saved network must match the order of the fits in the running ChiSurf instance. After loading a GraphML file containing the parameter links the values of the parameter and their connections are restored (**Fig.39**).

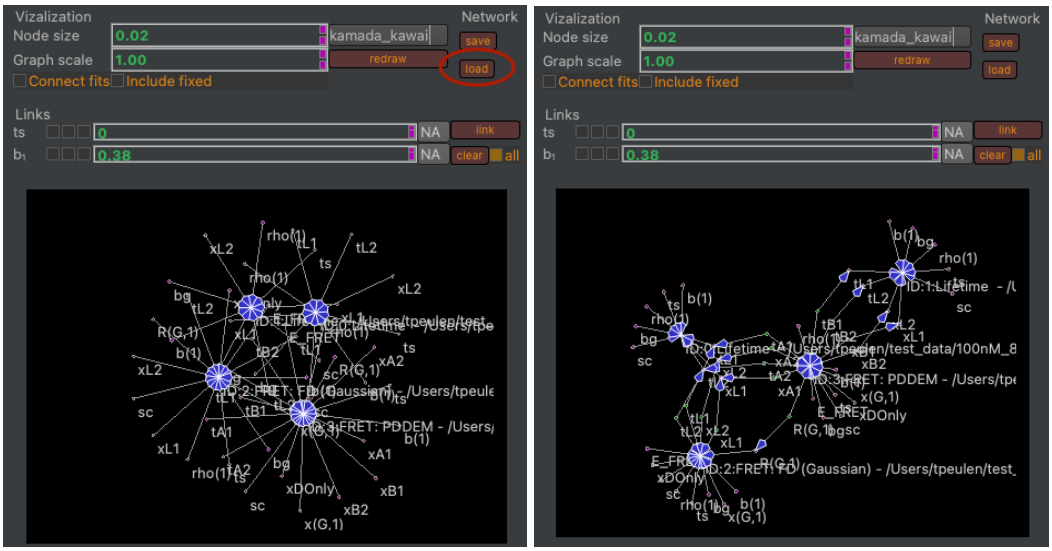


Fig.39. Loading parameter values and connections (left). After loading a matching GraphML file the dependencies between parameters are restored (right).

Linking parameters

Parameters can be selected by clicking on the respective nodes. Selected nodes are highlighted in magenta (**Fig.40**). The parameters of the selected nodes are displayed in the Links group box (**Fig.40**). The “link” button creates a link between selected nodes. The clear button removes a link between selected nodes. The “all” checkbox controls if clearing removes the link between selected nodes or the link between all nodes.

Fig.40. Selecting and linking parameters. The parameters “tB1” and “tL1” are selected (highlighted by magenta circles). Selected parameters are displayed in the Link group box. The clear button removes a link between selected parameters. Checking the “all” checkbox before clicking on clear removes all links. The “link” button creates a link between selected parameters.

7. Tutorials

7.1. Analysis of live-cell FCCS experiments

Computing Fluorescence Correlation Spectroscopy curves

In this tutorial, we will first work solely with the Fluorescence Correlation (FCS) curves exported from your calibration data and use the different measurement to characterize the daily performance of your system. It is assumed that you have computed FCS curves. In case you are still searching for a software, which fits your needs best, you might take a look at (https://fret.community/software). Once you have your workflow and templates established the calibration data is easily analyzed. The second part of this tutorial focuses on (*i*) adding your own fit models, and (*ii*) analyzing data of live-cell experiments.

Of course, all analysis can also be performed with other software, which supports the implementation of the described fit models and global analysis.

Data format

For analysis with ChiSurf, your correlation data has to be exported in text format with either three or four columns:

- Column 1: correlation time
- Column 2: correlation amplitude
- Column 3: first value reflects the measurement time, second value the average count rate, the rest of this column is filled with zeros
- Column 4: standard deviation of the correlation amplitude

The measurement time and the average count rate in column 3 are used to estimate the uncertainties in your correlation amplitudes if the standard deviation is not available, i.e. single measurement was performed or the hardware / software used for correlation does not provide these values.

Analysis of Calibration measurements

Provided test data - calibration

Sample	Purpose	Measured
ddH2O / buffer	background / dark counts	2x
IRF (EB quenched with KI)	Instrument response function	2x
A488	green calibration dye	5x
A568	red calibration dye	5x
DA-labeled DNA	green-red overlap calibration	5x

Settings:

- 2-color excitation (PIE)
- Excitation power: 485 nm: 6 µW, 568 nm: 1.4 µW
  - measured at objective
- 10 MHz rep rate for each laser line (485 nm prompt, 568 nm delay)
- TAC window: 100 ns (50 ns delay between pulses)
  - TCSPC bin size: 4 ps
- Calibration samples were measured as drop on objective slides

Provided test data – live cell experiments

Sample	Purpose	Description
eGFP	Donor-only sample (DOnly)	eGFP inserted into the intracellular loop 3 of the β_2 adrenergic receptor (β_2AR)
SNAP	Acceptor-only sample (AOnly)	SNAP-tag attached to the C-terminus of the β_2AR, labeled with SNAP Surface DY-549
NT-SNAP	Double-labeled sample (DA) which does not show FRET	β_2AR construct with both eGFP inserted into the intracellular loop 3 and SNAP-tag attached to the N-terminus, labeled with SNAP Surface DY-549

Settings:

- Experimental settings are identical to the calibration measurements.

Provided test data – simulation

Sample	Purpose	Description

CT-SNAP	Double-labeled sample (DA) which shows FRET	β_2AR construct with both eGFP inserted into the intracellular loop 3 and SNAP-tag attached to the C-terminus, labeled with SNAP Cell TMR

Settings / Assumptions:

- Bimodal diffusion of molecules with 30 % of a fast diffusing species (t_{D1} = 1 ms) and the rest of the time diffusing slowly with t_{D2} ~ 100 µs
- Dynamic exchange between two equally populated FRET states, high FRET (HF; FRET efficiency E = 0.7) and low FRET (LF, with E = 0.2), with a time constant of t_R = 70 µs
- Additionally, 16 % of triplet blinking at 5.5 µs was added.

Determination of background / dark count rate

Firstly, the average count rate from ddH2O or appropriate buffer (untransfected cells in case of cells) is obtained – under the same excitation condition as for the sample measurements!

Channel number	Channel name	Count rate [kHz] (prompt)	Count rate [kHz] (delay)
0	g-p (green parallel)	0.13	
1	r-p (red-parallel)	0.445	0.375
2	g-s (green perpendicular)	0.23	
3	r-p (red perpendicular)	0.79	0.39

The average countrate can be determined from your data in different ways:

- If you measured with the Picoquant software, then the average countrate is written in the header section and can be easily obtained by opening the dataset in Symphotime.
- Else you can get the number of photons detected during measurement time from an export of a photon arrival time histogram and divide this number by your measurement time
- Additionally, you can use the "determine_countrates.py" script to export this information for your whole folder of image (see Help file for using scripts).

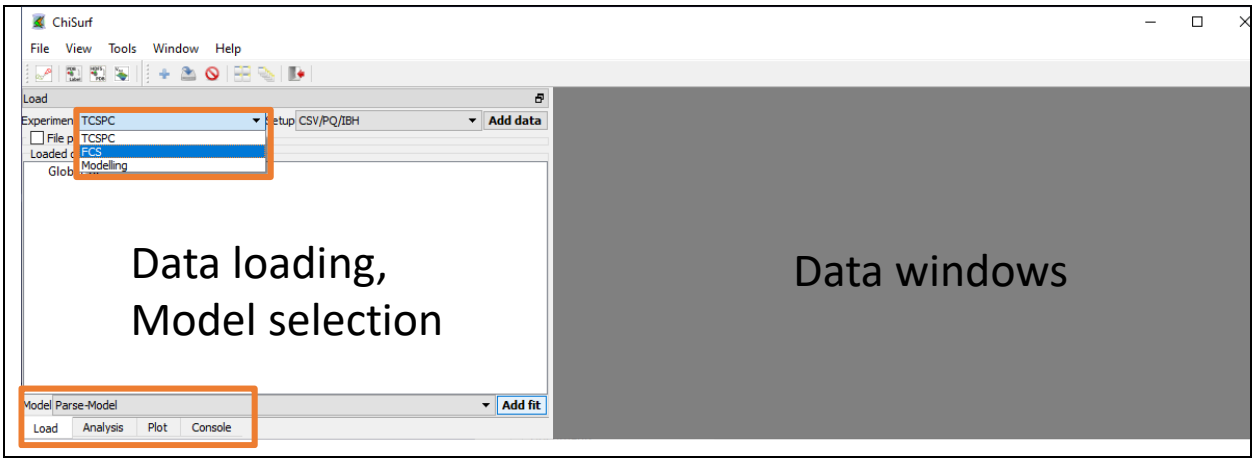
Calibration of detection volume – green excitation

In a first step, we will use the data from the individual diffusing red and green fluorophores and determine (i) the shape factor of the confocal detection volume, and next, based on this result and on the known diffusion coefficients, (ii) the confocal detection volume in femto liter. Additionally, the (iii) molecular brightness of your fluorophores and finally, (iv) the concentration of the samples can be calculated based on the fit results.

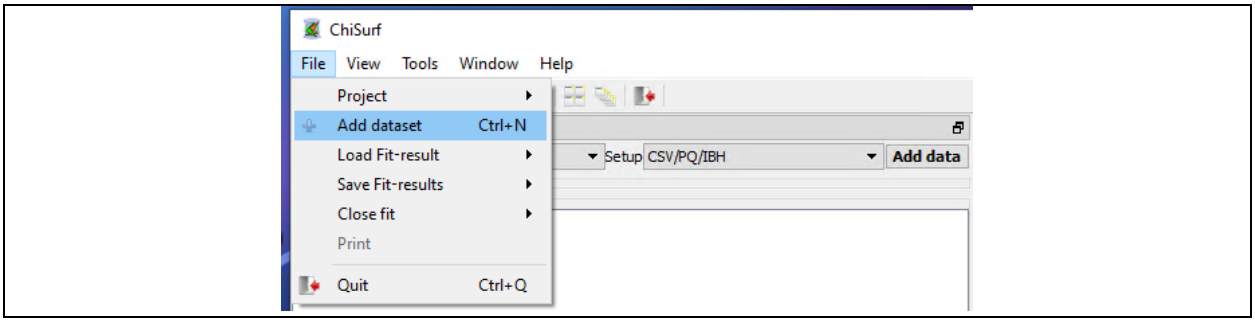
Fit of a FCS curve

➔ Open Chisurf2016
  - It might take a while to open – be patient!

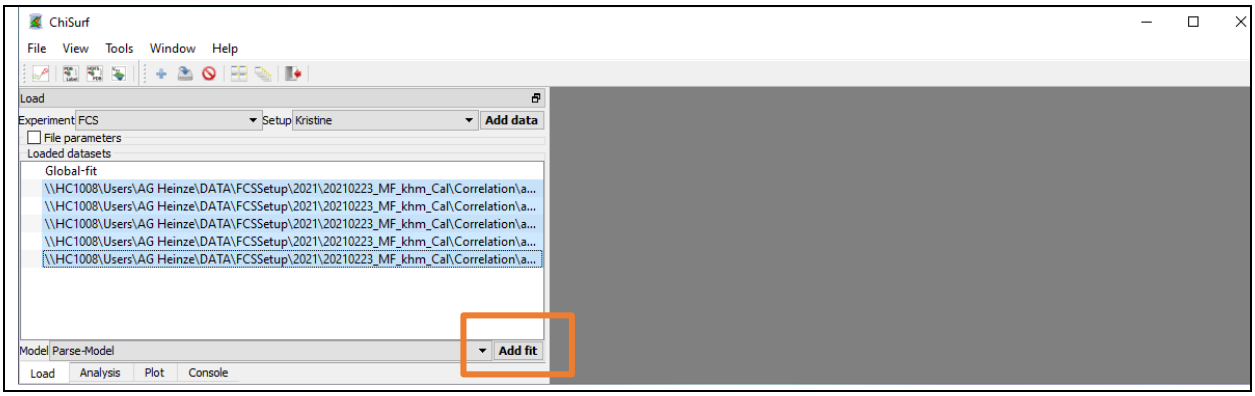
➔ It has got two panels:
  - Left / white: Here, the data is loaded and the analysis method / model is selected. Additionally, you can use the integrated Python-console to run little scripts.
  - Right / grey: Here, each data set opens in its own window.



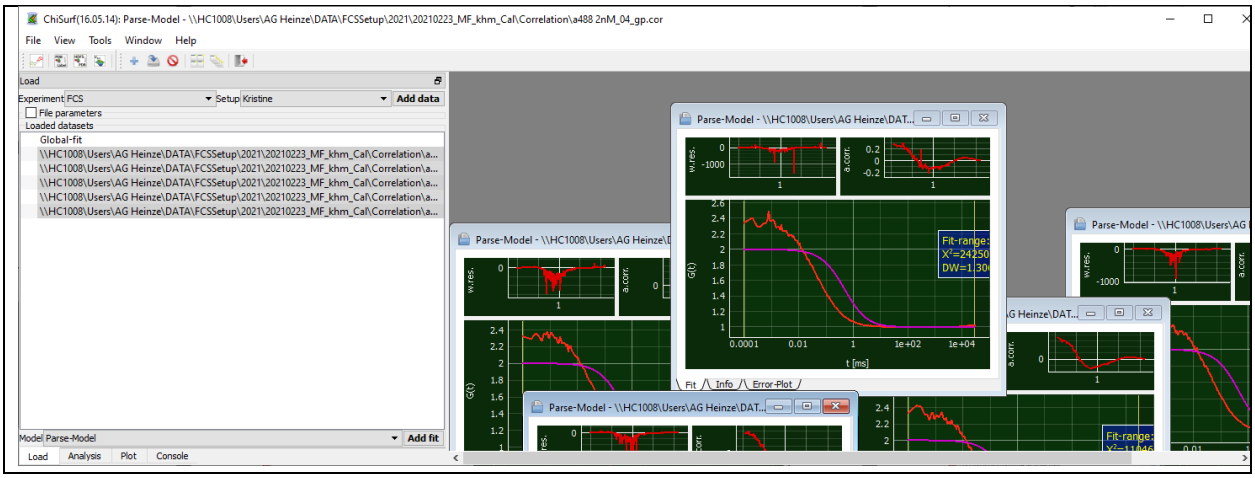
➔ Change to the "FCS" mode und load your all "***A488_ACF_prompt.cor***" (auto correlation curves of green calibration fluorophore correlated within the prompt time window) dataset(s) by "File" → "Add dataset"or simply by "drag'n'drop" into the white area.



➔ Select your loaded dataset(s) and click on "Add fit":



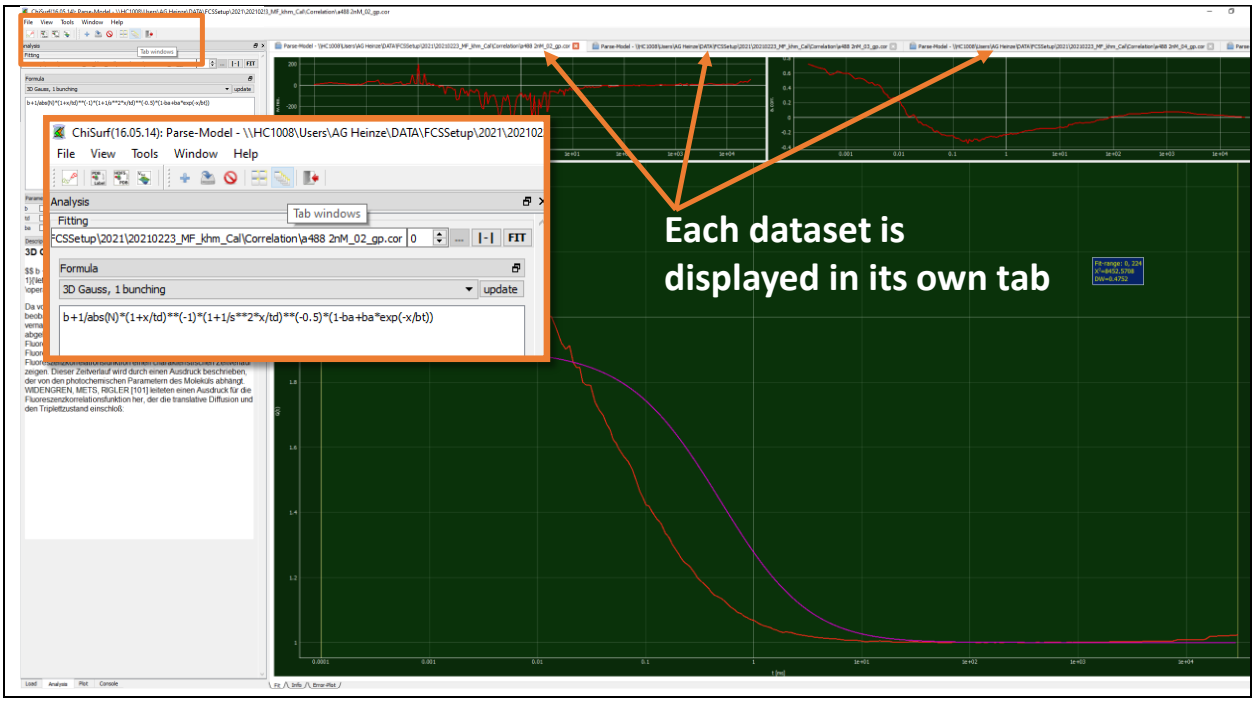
➔ Here, five new windows will open in the grey working panel:



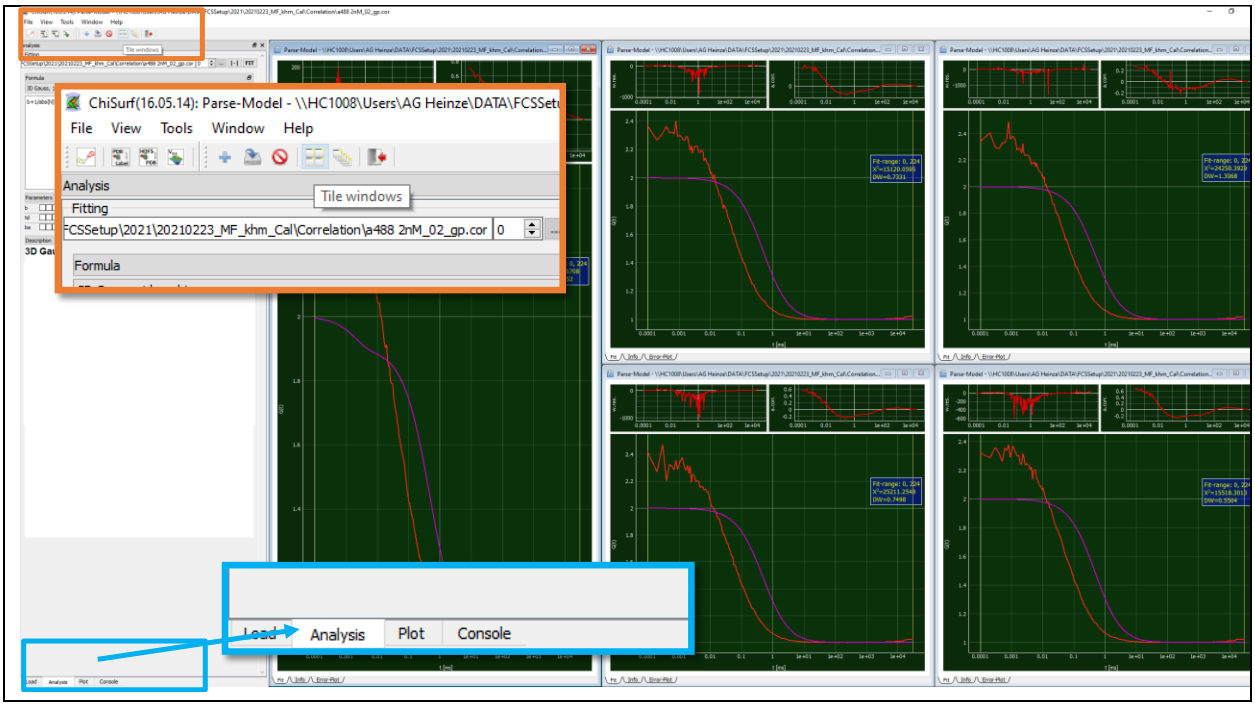
➔ Caution! The individual windows may lie on top of each other!

➔ Move the top-visible window aside and arrange the windows as you find it comfortable:

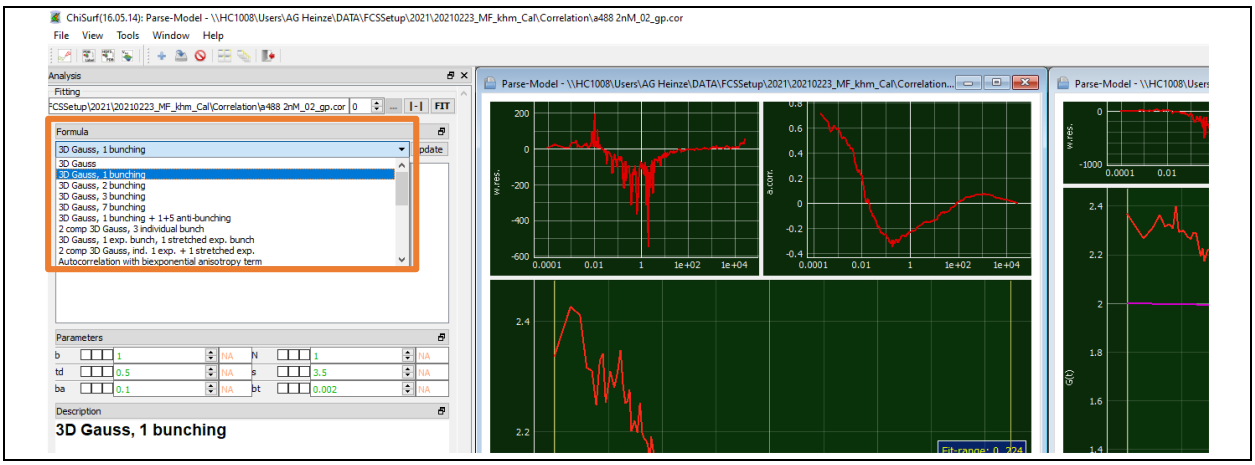
- You can maximize the windows to see only one curve at a time. Minimize the window to see the other data again.
- Alternatively, you can display each curve in full-size but in individual tabs. Then you can switch between the data by clicking on the respective tab and / or use the little arrows on the side to change between the tabs.



- Or you can distribute all windows automatically to evenly cover the full working panel:



➔ Now change to the analysis tab in the data panel and select the "3D Gauss 1 bunching" model from the drop-down menu for your first curve:

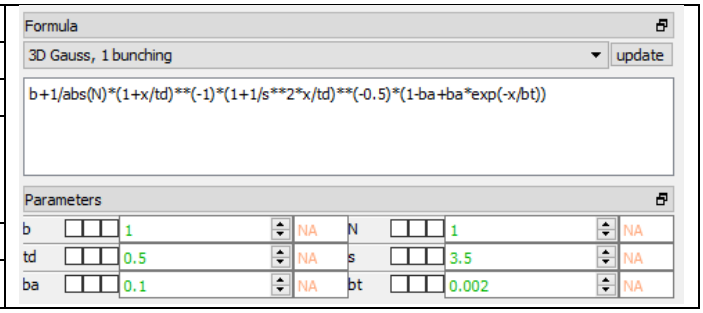


➔ This model has six parameters:

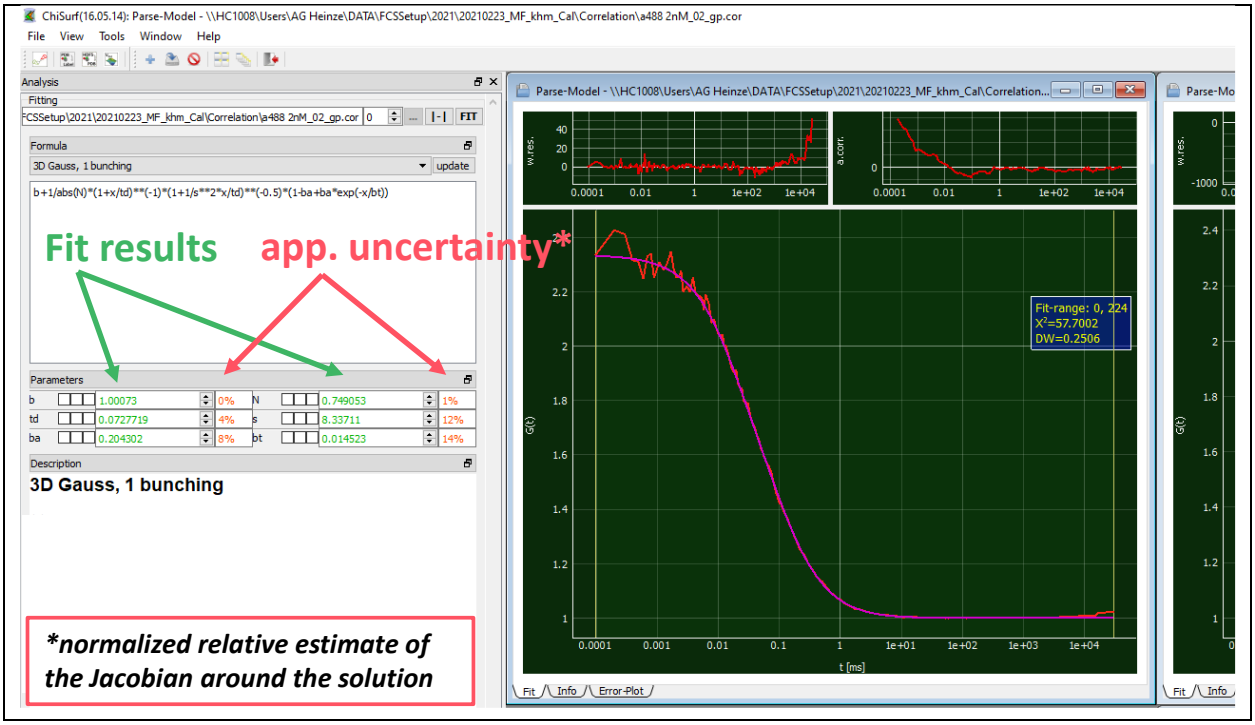
➔ The "1 Bunching" term as described by "ba" and "bt" is used to model the typical photo-physical triplet blinking of many fluorophores in the µs time range.

- Note: ChiSurf comes upon installation with a selection of pre-defined fit models, you can modify these fit models or add your own models easily.
- This will be shown later.

b	offset of the curve (either 1 or 0)
N	number of molecules in focus
td	diffusion time of the molecule [ms]
s	shape factor of the confocal volume: $s = z_0/w_0$ (ratio height / width)
ba	amplitude of relaxation time
bt	time constant of relaxation time [ms]



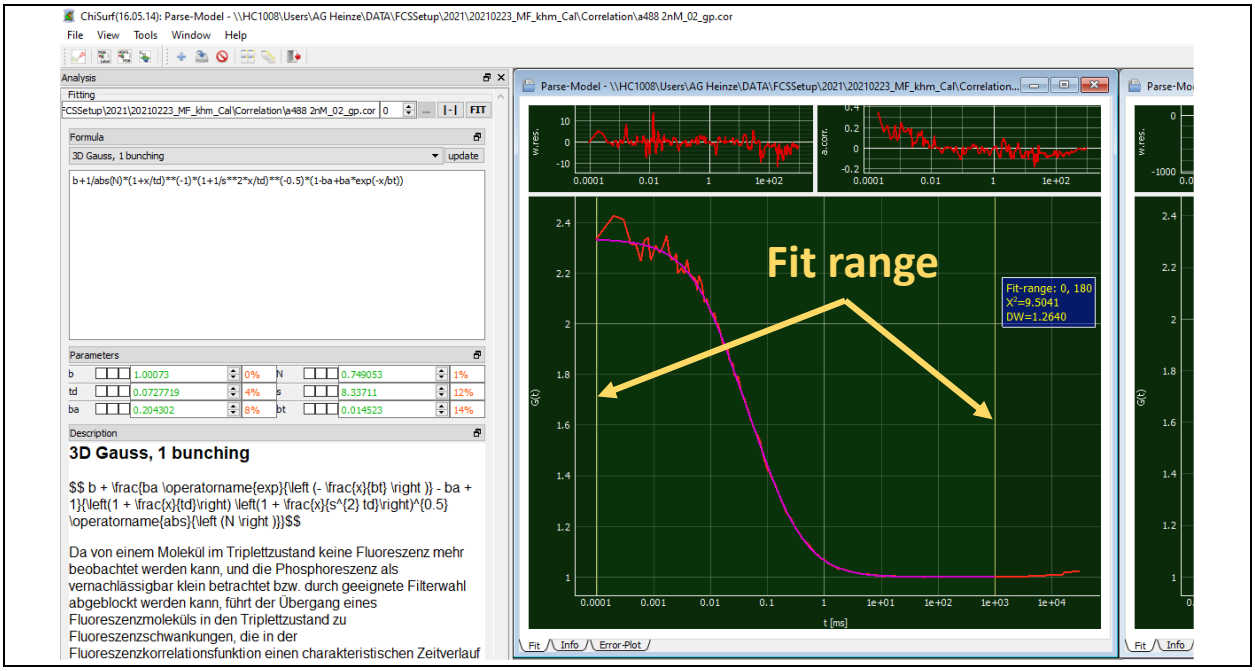
➔ Press "Fit" for fitting:



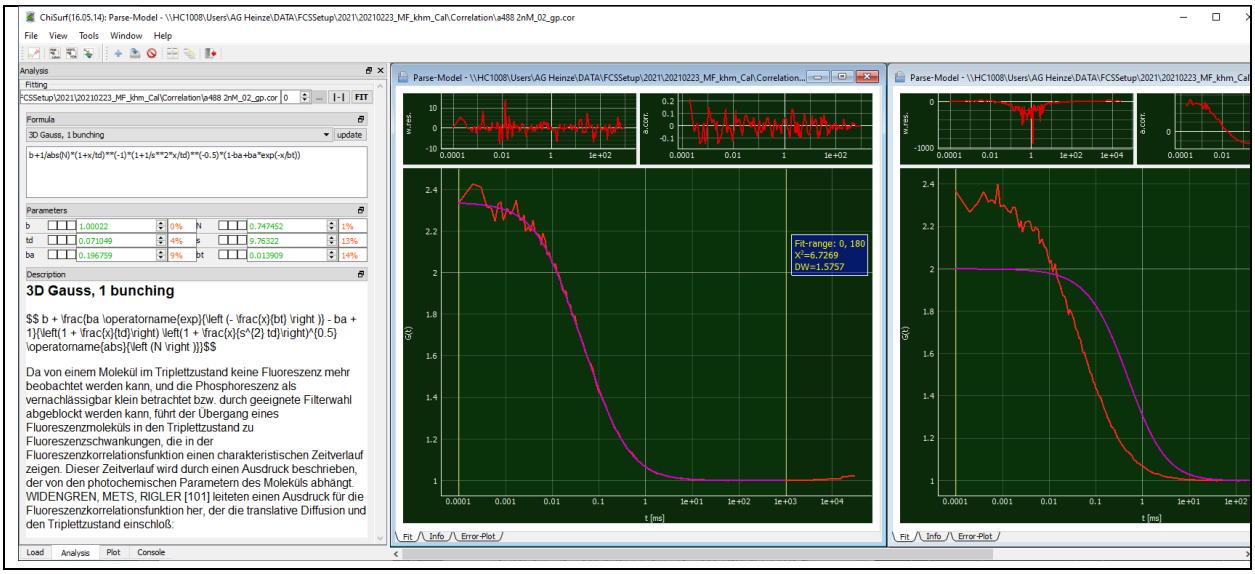
➔ We obtain a number of molecules in the focus N = 0.75, a diffusion time td = 0.073 ms (i.e. 73 µs) and triplet blinking time constant of 14.5 µs with an amplitude of 0.20. However, from the weighted residuals and the autocorrelation of the residuals, we can see a mismatch at long correlation times: This is because the absolute measurement time in this measurement was too short to reliable obtain these values.

➔ Next to the fit results, also the *normalized relative of the Jacobian Matrix around the solution* can be seen. This is **NOT** reflecting the **uncertainty** of the fit result, but the values might give a first hint whether the uncertainty is rather large or small. For more details on this topic, please check out the information provided on the following web page and the references cited herein:
https://root.cern.ch/doc/master/Minuit2Page.html

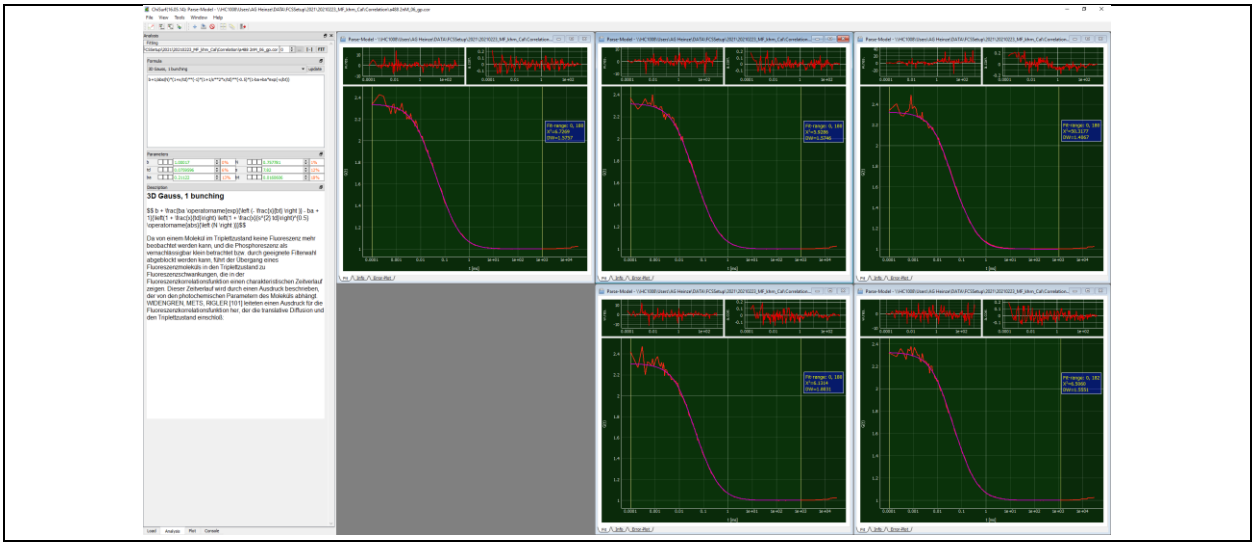
- ➔ For a more reliable estimate on the uncertainty of the fit parameter you have two options within ChiSurf to (i) sample the χ^2-surface or (ii) to run a Markov-Chain Monte-Carlo simulation, which also allows you to obtain the mutual dependencies between the fit parameter. However, this uncertainty analysis is beyond the scope of this analysis of the calibration samples and will be shown in a different tutorial.
- ➔ Here, we take advantage of multiple measurements of the same sample and take these as additional restraints.
- ➔ Shorten the fit range to ~ 100 ms by grabbing the right yellow line and move it to the left:



- ➔ *Of note*: To reliably fit your diffusion time, the **baseline** (0 or 1, depends on correlation algorithm) **MUST be reached** in your correlation curve (or in the fit range, respectively)
- ➔ Press “fit” and observe the changes:

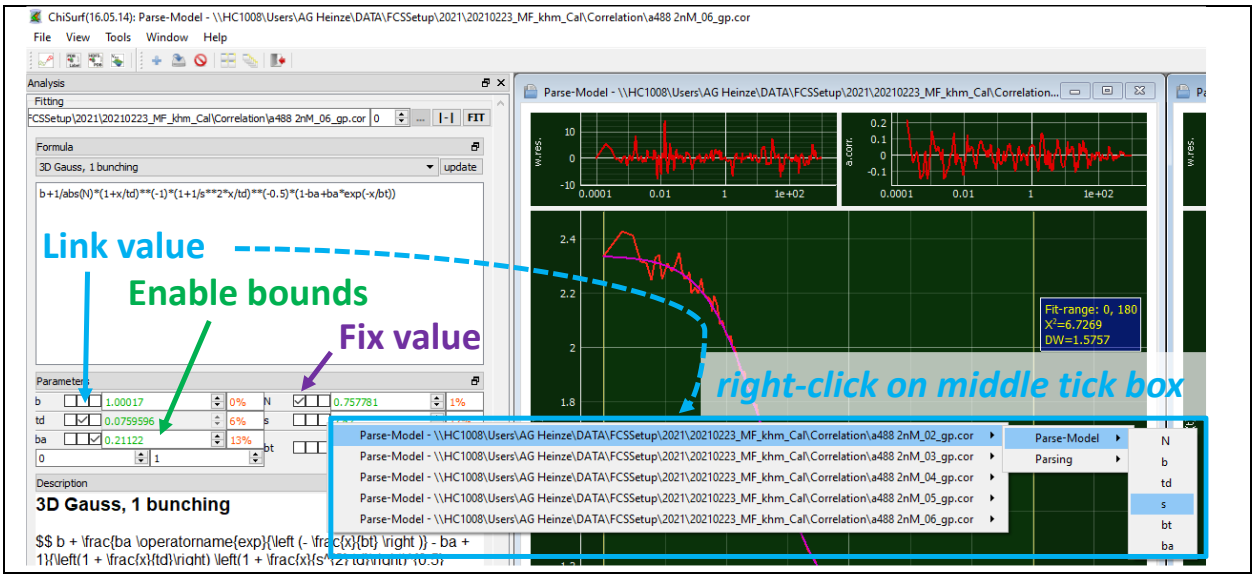


- ➔ Now the shape has increased to 9.76, which is very huge and would indicate a misalignment of your system. In ideal case, the shape factor should lie between 3 - 7. The other values have changed only slightly.
- ➔ Now let's add the other measurements into the play and see whether a global fit of all measurements stabilizes this value.
- ➔ Go to the other fit windows, change the fit to "3D Gauss, 1 bunching", adjust the fit range and fit them as done for the first curve:



- ➔ Next, we will link the fits together such that the fit parameter are jointly minimized
- ➔ For this, first decide for one "parent" dataset, to which all other datasets are pointing. Here, I will simply take curve #1.

➔ Now, switch to the first of your "child" or "dependent" dataset. We will now work with the three checkboxes located between each variable name and variable value:

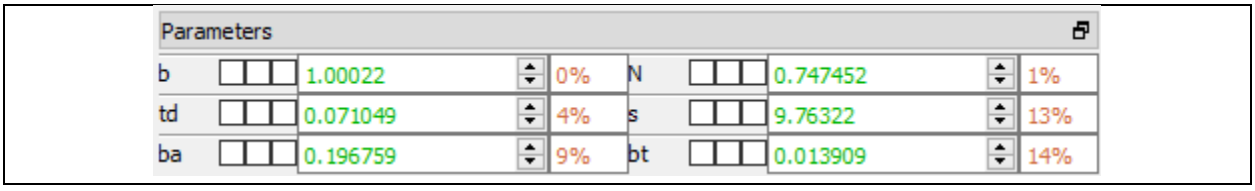


➔ Each tick box has a different function:

- Left: fixes the value of the parameter to its current value
- Right: Two new parameter fields open, in these ones the lower (left) and upper (right) boundaries for this parameter during the fitting can be defined. An example is to define the allowed fit range for a correlation amplitude to be positive and lie between 0 -1.
- Middle: By right-clicking into this tick box a list of all opened fit windows opens. Move your mouse towards the right, as soon as you approach the little arrow, new options appear, move further right on the height of "Parse Model" until the list of fit parameter appears.

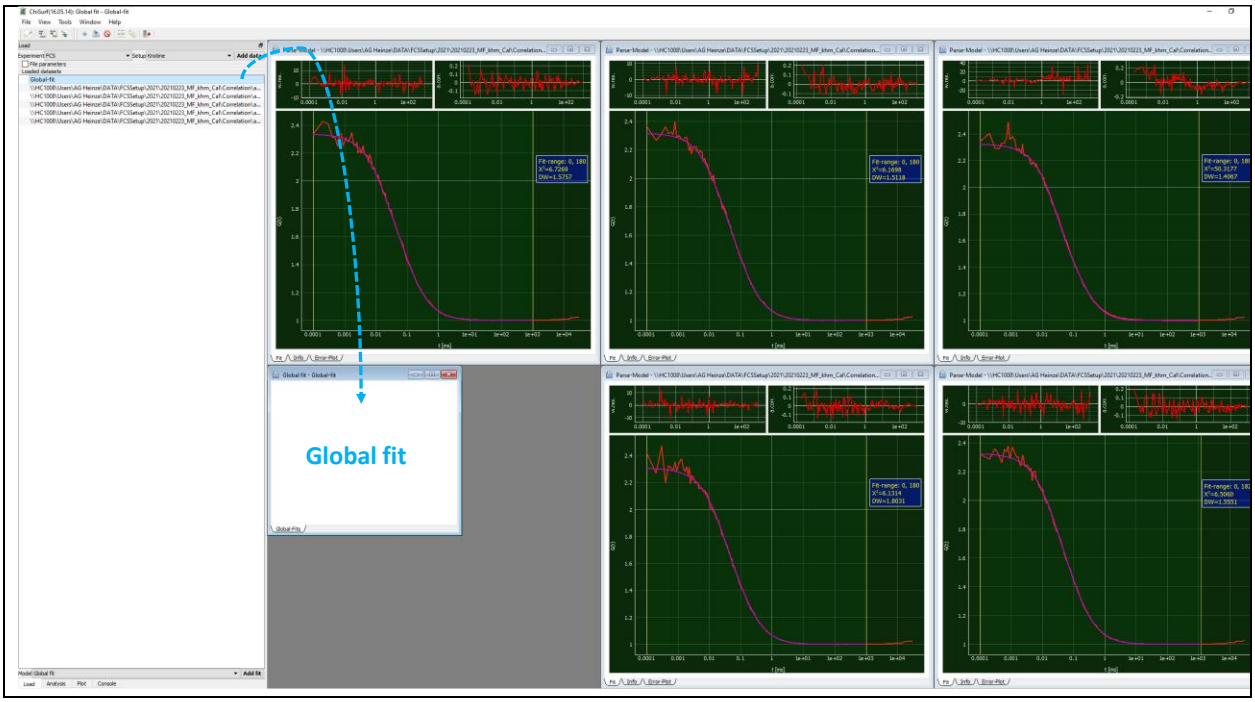
➔ From this list of fit parameter select the appropriate one.

➔ For our global fit, we will now link (i) the diffusion time td, (ii) the shape parameter s and (iii) the triplet time constant bt to the first data set:

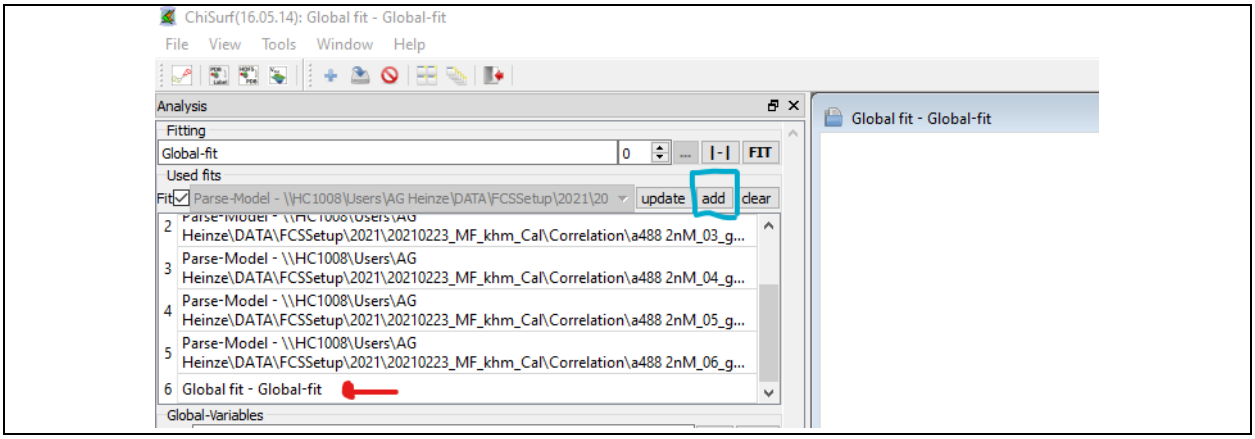


➔ Linked parameter will appear greyed out.

➔ After you are done with the linking, add a new global fit in the "load" tab:



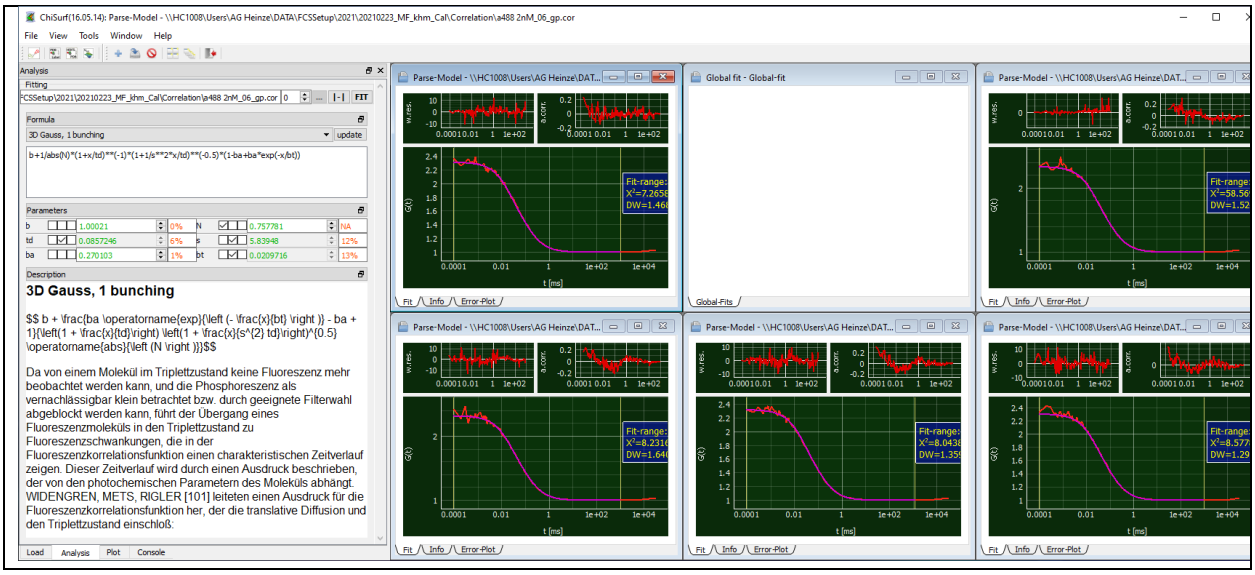
- ➔ It opens as empty / white window.
- ➔ Switch back to the analysis tab and add dataset which are to be jointly fitted by clicking firstly on “update” and then on “add”.



- ➔ Now a list of all available fit windows appears.
- ➔ If you want to select which datasets to add, remove the tick behind “fits”. Then you can choose from a drop-drown list which datasets are to be included in the global fit.
- ➔ We need to remove the “global fit” from list (“circular reference”). You do so by simply double-clicking on the item in the list.
- ➔ Now, press fit and observe what happens.
- ➔ If you fit many and / or complicated models, the program might take a few moments and display “not responding”. This is nothing to worry about and wait until it responds.

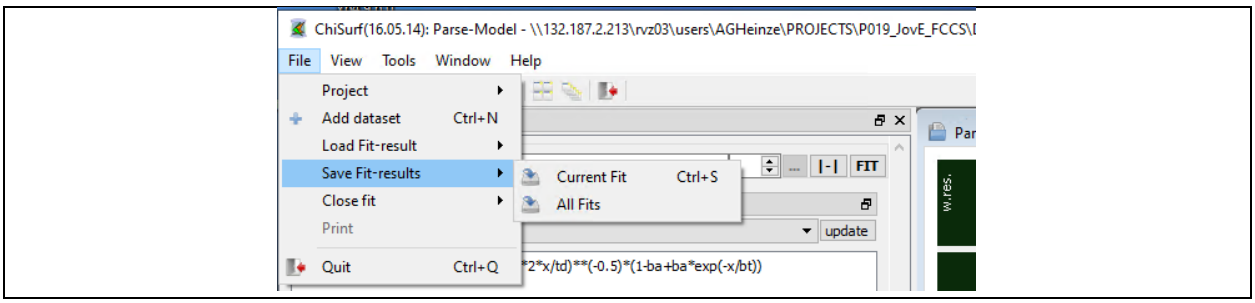
➔ Now all datasets have fitted jointly and we obtain the following values:

- td = 85.7 µs
- s = 5.84
- bt = 21.0 µs
- ba varies between 0.255 – 0.270
- N varies between 0.748 - 0.766



➔ Finally, let's save the fits by either selecting "File" → "Save Fit-results" → "current fit" or by pressing "Ctrl + S".

➔ Caution! Saving all fits may not work, if the filenames are (a) similar and (b) the whole file path is too long. (AutoSaving uses complete path as automatic save name currently).



➔ Save the results of all measurements, we will need the fit results in the next step.

Calculation of confocal volume

Note:

- Use the provided excel sheet "*FCCS_calibration.xlsx*" to retrieve the results semi-automatically.

- Don't forget to update each time your fit results from the respective measurement day!
- Please note the different units (µm, m etc.) and unit conversions!

➔ To determine the confocal volume, we need three different values:
  - Diffusion coefficient of the freely diffusing standard dye, here A488,
    - These values can be found in the literature for most common fluorophores.
    - Caution! They are temperature and solvent dependent, i.e. if you measure at low temperature or in a more viscous environment, you have to correct for the viscosity of the solution.
    - **D_{A488} = 414 µm²/s** (@25°C in ddH2O [2]
  - Diffusion time t_D from our fit: **85.7 µs**
  - Shape factor ***s*** from our fit: **5.84**

➔ Diffusion time and diffusion coefficient are related by the following relationship:

$$D = \frac{w_0^2}{4t_D} \quad \rightarrow \quad w_0^2 = 4t_D D$$

➔ Inserting our fit results and *D* into this equation, we obtain **w_0 = 0.377 µm**

➔ Based on the value from w_0, we can determine the height z_0 of the confocal volume:

$$s = \frac{z_0}{w_0} \quad \rightarrow \quad z_0 = s\, w_0$$

➔ For our example, **z_0 = 2.2 µm**

➔ Finally, the confocal volume is assumed to have in good approximation an elliptical shape from which the volume V_{eff} can be obtained using the following formula:

$$V = \pi^{3/2} z_0 w_0^2$$

➔ Thus, our confocal volume in the green excitation range has a size of **$V_{eff,green}$ = 1.74 fL**.

Estimation of the fluorophore concentration

➔ Using the above determined confocal volume, the concentration of the fluorophore in your measurement solution can be estimated.

➔ The following parameters are required:
  - Confocal volume **$V_{eff,green}$ = 1.74 fL**
  - Number of molecules in focus, ***N***, from our fit: **0.76** (in average)
  - Avogadro's number: **N_A = 6.022*10^{23} mol^{-1}**

➔ The concentration of the fluorophore is determined from the following relationship:

$$c = \frac{N}{V_{eff,green} N_A}$$

➔ The approximated concentration of green calibration dye is calculated to be ***c_{A488}*** ~ **0.72 nM.**

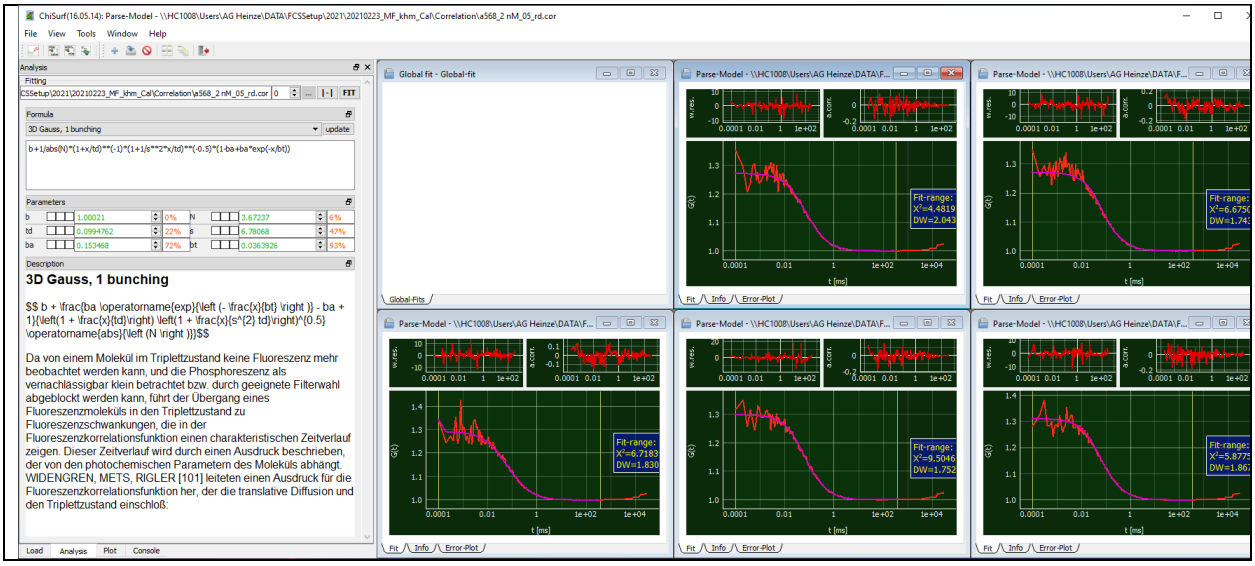
Confocal volume			
D	414	µm²/s	*Literature value*
	4.14E-10	m²/s	
td	85.7	µs	*Fit result - please add!*
	0.0000857	s	
s	5.84		*Fit result - please add!*
w02	1.41919E-13		
w0	3.76722E-07	m	
	0.377	µm	
z0	2.20005E-06	m	
	2.200	µm	
pi^(3/2)	5.568327997		
V	1.7386E-18	m³	
	1.739	fl	1 fl = 1e-15 l

Concentration estimation			
Alexa488			
N	0.76	number	*Fit result - please add!*
c	4.34833E+14	number/liter	
c	7.22E-10	Mol/l	
	7.22E-07	mM	
	7.22E-04	µM	
	0.72	nM	

Calibration of detection volume – red excitation

➔ For the red detection volume, we proceed as described for the green detection volume.
➔ Only the fit results will be provided for a guidance.
➔ Important: Use the correlated signal of the directly excited red fluorophore within the delay time window of your measurement, e. g. "***A568_ACF_delay.cor***"

Fit of FCS curve



➔ Here, also the "3D Gauss, 1 Bunching" model will be sufficient.
➔ We obtain:

- td = 99.5 µs
- s = 6.78
- N varies between 3.16 – 3.67

➔ Save the fit results and proceed with the calculations as described for the green detection volume.

Calculations

➔ As A568 is larger than A488, its diffusion coefficient is reduced compared to A488.
➔ There are different studies reporting a D_{A568} in the range of 330 -360 µm²/s.
➔ Here we use a value of **D_{A568} = 363 µm²/s** [3].

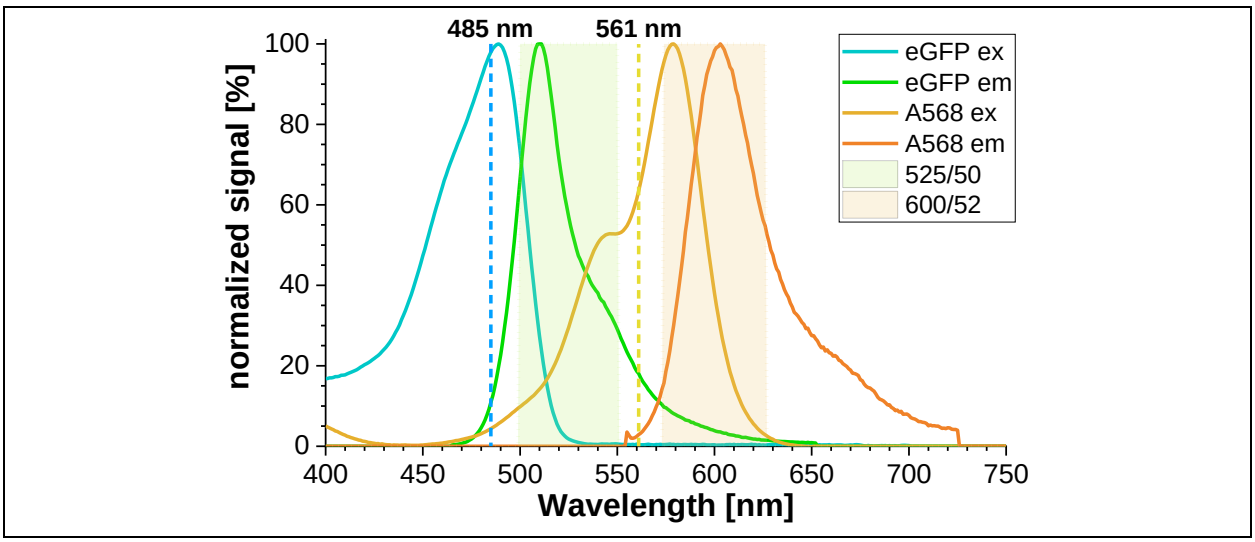
Confocal volume			
D	345	µm²/s	*Literature value*
	3.45E-10	m²/s	
td	99.5	µs	*Fit result - please add!*
	0.0000995	s	
s	6.78		*Fit result - please add!*
w02	1.3731E-13		
w0	3.70554E-07	m	
	0.371	µm	
z0	2.51235E-06	m	
	2.512	µm	
pi^(3/2)	5.568327997		
V	1.92091E-18	m³	
	1.921	fl	1 fl = 1e-15 l

Concentration estimation			
Alexa488			
N	3.44	number	*Fit result - please add!*
c	1.7929E+15	number/liter	
c	2.98E-09	Mol/l	
	2.98E-06	mM	
	2.98E-03	µM	
	2.98	nM	

➔ As expected for a measurement at red-shifted, thus longer wavelengths of excitation and emission, the confocal detection $V_{eff,red}$ volume is increased compared to $V_{eff,green}$.

Correction factors for spectral crosstalk & direct acceptor excitation

➔ Absorption and emission spectra of fluorophores are usually quite broad and extent over quite a wavelength range, and not seldom the spectra from green and red fluorophores used in the measurement overlap.
➔ By introducing optical (bandpass) filters into the emission path of your setup, it is possible to minimize these effects.



➔ However, even if these contributions to your signal are small, they might be become critical when you have a low count rate / low signal in one channel compared to the other channels.

➔ For more detailed information on required corrections and the nomenclature of correction factors, we refer to the recently published multi-laboratory benchmark study of Hellenkamp *et al.* [4].

Spectral crosstalk of green fluorescence into red detection channel

➔ Required is the measurement from the freely diffusing green fluorophore.

➔ Collect the count rates in all four detection channels for the ddH2O /buffer measurement and the green dye (A488) measurement:

Channel number	Count rate [kHz] BG	Count rate [kHz] A488	Corrected count rate [Hz]
0	0.13	5.30	5.18
1	0.36	1.23	0.87
2	0.23	5.38	5.15
3	0.69	1.53	0.85

(Note: average values of repeated measurements are reported)

➔ The background count rate needs to be subtracted from value and then the ratio of count rates in the red channels to the green channel is determined:

$$\alpha = \frac{\left(\left(CR_{1,A488} - CR_{1,BG}\right) + \left(CR_{3,A488} - CR_{3,BG}\right)\right)}{\left(\left(CR_{0,A488} - CR_{0,BG}\right) + \left(CR_{2,A488} - CR_{2,BG}\right)\right)} * 100\%$$

➔ In our example the spectral crosstalk ***α* = 16.6 %.**

Determination of direct excitation of red fluorophore by green excitation

- ➔ Required is the measurement of the freely diffusing red fluorophore.
- ➔ Collect the count rates in the two red detection channels for the ddH2O / buffer measurement and the red dye measurement A568, take care to separate the count rate into the "prompt" and "delay" time window:

Channel number	Count rate [kHz] BG	Count rate [kHz] A568	Corrected count rate [Hz]
1 delay	0.36	4.24	3.88
1 prompt	0.36	1.90	1.54
3 delay	0.69	4.81	4.12
3 prompt	0.69	2.37	1.68

(Note: average values of repeated measurements are reported)

- ➔ In the "prompt" time window the green excitation pulse might excite a bit of the red fluorophores.
- ➔ In the "delay" time window the red fluorophores are directly excited.
- ➔ Subtract the background counts from each channel and the respective time window.
- ➔ Calculate the ratio of the "prompt" time window to the "delay" time window:

$$\delta = \frac{\left(\left(CR_{1,prompt,A568} - CR_{1,prompt,BG}\right) + \left(CR_{3,prompt,A568} - CR_{3,prompt,BG}\right)\right)}{\left(\left(CR_{1,delay,A568} - CR_{1,delay,BG}\right) + \left(CR_{3,delay,A568} - CR_{3,delay,BG}\right)\right)} * 100\%$$

- ➔ Caution! If you measure in a mixture of green and red fluorophores, you need to also subtract the green crosstalk into the red channels (α determined above!)
- ➔ For the shown example, the direct excitation of acceptor by green laser is **δ = 40.3 %.**

Determination of molecular brightness

- ➔ After having correcting the count rates, we can now determine the molecular brightness of our fluorophores.
- ➔ Note: It is advisable to monitor this molecular brightness of your calibration standards carefully as they can give you early hints about a possible misalignment and / or reduction of laser output power.
- ➔ Required parameter:
  - Number of molecules in focus as determined from the fits above
    - N_{green}: 0.76
    - N_{red}: 3.44
  - Corrected count rates:
    - $CR_{corr,green,0}$ = 5.18 kHz
    - $CR_{corr,green,2}$ = 5.15 kHz
    - $CR_{corr,red,1}$ = 3.88 kHz

- $CR_{corr,red,3}$ = 4.12 kHz

➔ The molecular brightness is calculated as count per molecule and second:

$$B_{green} = \frac{CR_{corr,green,0} + CR_{corr,green,2}}{N_{green}}$$

$$B_{red} = \frac{CR_{corr,red,1} + CR_{corr,red,3}}{N_{red}}$$

➔ For the data shown here, the following molecular brightness is obtained:

Countrates A488

	ch 0 / s [kHz]	ch 2 / p [kHz]	sum [kHz]	N	brightness [kHz/molecule]
2	5.27	5.38	10.65	0.766	12.28
3	5.26	5.34	10.60	0.753	12.44
4	5.30	5.36	10.66	0.757	12.50
5	5.30	5.39	10.69	0.748	12.66
6	5.35	5.43	10.78	0.756	12.63
Mean	5.30	5.38	10.68	0.76	12.50
stdev	0.03	0.03	0.06	0.01	0.14

Countrate background / **Countrate red channels**

	ch 0 / s [kHz] ddH2O	ch 2 / p [kHz] ddH2O	ch 1 / s [kHz] A488	ch 3 / p [kHz] A488
1	0.13	0.23	1.25	1.57
2	0.13	0.23	1.23	1.54
3			1.20	1.52
4			1.22	1.52
5			1.23	1.53
Mean	0.13	0.23	1.23	1.53
stdev	0.00	0.00	0.02	0.02

Green crosstalk into red channel

	CR [kHz] A488	CR [kHz] BG	CR_{A488}-CR_{BG} [kHz]	
0 prompt	5.30	0.13	5.18	*spectral crosstalk [%] α*
1 prompt	1.23	0.36	0.87	16.61
2 prompt	5.38	0.23	5.15	
3 prompt	1.53	0.69	0.85	

Countrates A568

	ch 1 / s [kHz] delay	ch 3 / p [kHz] delay	sum [kHz]	N	brightness [kHz/molecule]
1	4.55	5.11	9.66	3.16	3.06
2	4.43	5.00	9.43	3.30	2.86
3	4.25	4.83	9.08	3.50	2.60
4	4.18	4.74	8.92	3.59	2.48
5	4.10	4.66	8.76	3.67	2.39
Mean	4.24	4.81	9.17	3.44	2.68
stdev	0.12	0.13	0.33	0.19	0.25

Countrate background

	ch 1 / s [kHz] ddH2O	ch 3 / p [kHz] ddH2O	ch 1 / s [kHz] prompt	ch 3 / p [kHz] prompt
1	0.390	0.69	2.01	2.46
2	0.36	0.69	1.95	2.41
3			1.88	2.35
4			1.85	2.33
5			1.83	2.29
Mean	0.36	0.69	1.90	2.37
stdev	0.00	0.00	0.07	0.06

Direct excitation of A568 by 488-laser line

	CR [kHz] A568	CR [kHz] BG	CR_{A568}-CR_{BG} [kHz]	
1 Delay	4.24	0.36	3.88	*direct excitation [%] δ*
1 Prompt	1.90	0.36	1.54	40.31
1 Delay	4.81	0.69	4.12	
1 Prompt	2.37	0.69	1.68	

Determination of overlap of green & red detection volume

➔ For calibration of the overlap of the green and red excitation volume, DNA strands are used, which carry both the green and red calibration fluorophore. However, the fluorophores are placed usually so far apart that no energy transfer due to FRET can occur.

➔ For single-molecule measurements, a mixture of DOnly-labeled DNA strands and high- and low-FRET showing DNA strands is additionally used to calibrate the green-to-red detection efficiency ratio.

➔ For our determination of the confocal volume overlap, we will use the amplitude of the green (ACF_{green}) and red autocorrelation function (ACF_{red}) as well as from the "Green-prompt"-"Red-delay" cross-correlation function (CCF_{PIE}).

Important note!

The detection volume increases with the excitation and emission wavelength of the fluorophores. This effect is more prominent in diffraction-limited setup as used here and commonly for live cell FCS.

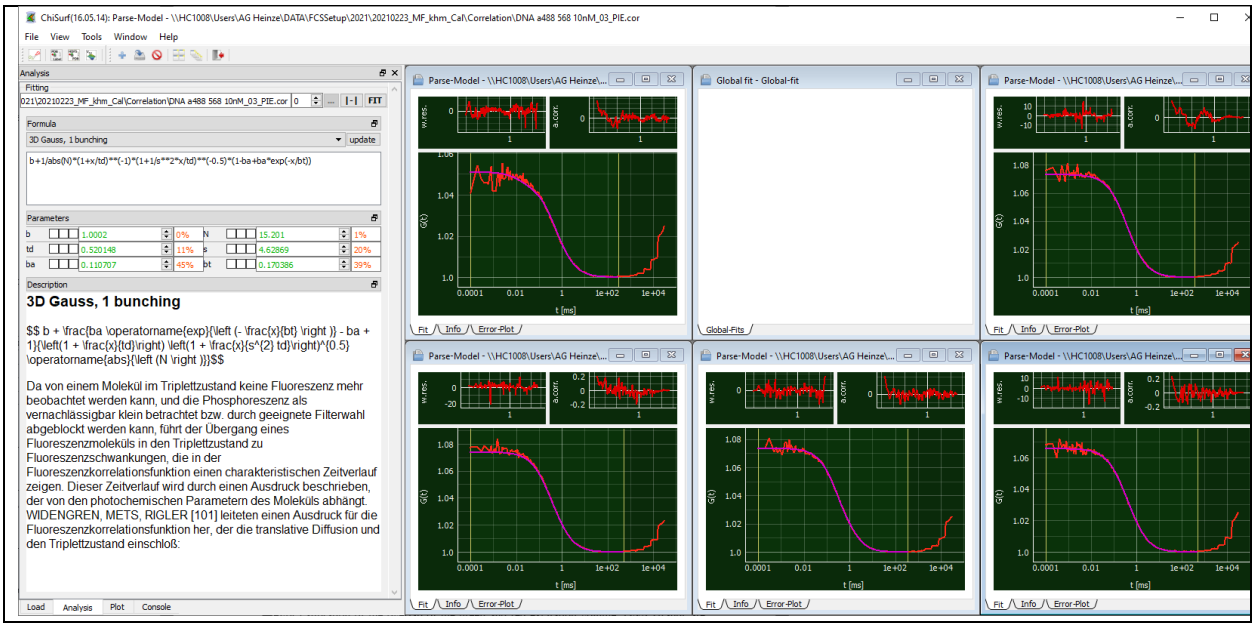
In setups for single-molecule experiments, with freely diffusing molecules, with larger detection volumes this effect can often be ignored and the diffusion terms in the green and red channels can be fit jointly.

Global fit of FCS curves

- ➔ Switch back to ChiSurf2016 and load the following correlation curves form your DNA-sample:
  - *DNA_gp.cor*: Autocrrelation of green channels in prompt time window
  - *DNA_rd.cor*: Autocorrelation of red channels in delay time window
  - *DNA_PIE.cor*: Crosscorrelation of green signal in the prompt time window with red signal in the delay time window
- ➔ Add a "3D Gauss, 1 bunching" model to your correlation functions
  - Of note: theoretically no bunching term should be required as the photophysics from the two fluorophores is independent from each other, and is thus not resulting in a correlating signal.
  - However, in practice, in case of significant photophysics / triplet blinking and considerable crosstalk as is the case here, we often observed an apparent "photophysics" term, which we model by the relaxation term.
- ➔ Fix the shape factor s_{green} and s_{red} to your determined values from the free dye measurements above.
- ➔ Remember to adjust your fit range at long lag times if required.
- ➔ Note: If you have multiple DNA measurements, fit the respective correlation curve from the same correlation channels jointly, i.e. link ***td***, and ***bt*** and – for the CCF_{PIE} also s_{PIE}
- ➔ Observe s_{PIE} in your CCF_{PIE}, it should get a "reasonable" number, usually between s_{green} and s_{red}:
- ➔ Save all fit results, we will need them for our calculations below.
- ➔ Here, we obtain the following fit results for our DNA samples:

Parameter	DNA_gp	DNA_rd	DNA_PIE	
s*	5.7*	6.78*	4.63	
***td* [µs]**	438	474	520	
N	11.6 – 7.54	14.4 – 8.73	1/G(tc)	20.4 – 13.4
***bt* [µs]**	88.6	101	170	
ba	5 - 22 %	1.3 – 24.5 %	0 – 15.7	

**s_{green} and s_{red} are fixed from the calibration measurements above.*



Calculations

From our double-labeled DNA measurements, we need to derive two important parameter:

- The size of the overlapping confocal detection volume $V_{eff,PIE}$
- The Cross-correlation amplitude, which reflects 100 % co-diffusion

The size of the overlapping confocal detection volume can be determined using the already provided equations used in the sections above for the calculations of the confocal detection volume of the green and red channel.

In a first step, we use the obtained diffusion times from our *DNA_gp* and *DNA_rd* fits and determine the translational diffusion coefficient of our DNA sample:

$$D_{DNA,green} = \frac{w_0^2}{4t_{D,green}} \quad \text{and} \quad D_{DNA,red} = \frac{w_0^2}{4t_{D,red}}$$

Here, we obtain a value of $\mathbf{D_{DNA,green} = 81.8\ \mu m^2/s}$ and $\mathbf{D_{DNA,red} = 72.4\ \mu m^2/s}$ using the value of w_0 from A488 and A568 dye, respectively. Thus, in average $\mathbf{D_{DNA} = 77.1\ \mu m^2/s}$.

Based on this value, we can obtain $w_{0,PIE}$ and $z_{0,PIE}$:

$$w_0^2 = 4t_D D \quad \text{and} \quad z_0 = s\, w_0$$

Here, $\mathbf{w_{0,PIE} = 400\ nm}$ and $\mathbf{z_{0,PIE} = 1.85\ \mu m}$. This results in a $\mathbf{V_{eff,PIE}}$ of **1.66 fL**:

$$V = \pi^{3/2} z_0 w_0^2$$

Next, we observe the amplitudes of the auto- and cross correlation functions: In an ideal system the amplitudes of the three curves, *DNA_gp*, *DNA_rd* and *DNA_PIE* should be identical. However, as the detection volumes differ with the excitation and emission wavelength, this is rarely the case. In the next-optimal setting, the amplitude of *DNA_PIE* would be identical to the amplitude of the autocorrelation curve with the lower amplitude.
In common experimental settings, the overlap of the green and red confocal detection volumes is suboptimal and the apparent amplitude of a 100 % co-diffusion sample is required for calibration. The concentration, and thus, later the fraction of co-diffusing particles in your sample, of double-labeled particles can be calculated based on the ratio of the correlation amplitudes:

$$c_{RG} = \frac{G_{0,CCF}}{G_{0,ACFgreen}} \cdot c_{green} \qquad \text{and} \qquad c_{GR} = \frac{G_{0,CCF}}{G_{0,ACFred}} \cdot c_{red}$$

where the amplitudes $G_{0,ACFgreen}$ and $G_{0ACF,red}$ are the inverse of the respective number of particles, N_{green} and N_{red}, in focus.
Here, we obtain amplitude ratios for 100 % co-diffusion of ***ratio*$_{GR}$ = 0.57** for the green and of ***ratio*$_{RG}$ = 0.68** for the red autocorrelation curves.

Effective overlapping confocal volume			
D	77.09	µm²/s	based on green /red
	7.7086E-11	m²/s	
td	520	µs	Fit result - please add!
	0.00052	s	
s	4.63		Fit result - please add!
w02	1.6034E-13		
w0	4.0042E-07	m	
	0.400	µm	
z0	1.854E-06	m	
	1.854	µm	
pi^(3/2)	5.568328		
V	1.6552E-18	m³	
	1.655	fl	1 fl = 1e-15 l
			1000 l = 1 m³

	Green				Red			
	N	c [nM]	td [µs]*	D [µm²/s]	N	c [nM]	td [µs]*	D [µm²/s]
2	11.6	11.08	434	81.75	14.7	12.71	474	72.42
3	8.53	8.15	434	81.75	10.8	9.34	474	72.42
4	7.79	7.44	434	81.75	8.73	7.55	474	72.42
5	7.54	7.20	434	81.75	9.01	7.79	474	72.42
6	7.61	7.27	434	81.75	8.94	7.73	474	72.42
Mean	7.87	7.52	434.00	81.75	10.44	9.02	474.00	72.42
stdev	0.39	0.38	0.00	0.00	2.26	1.95	0.00	0.00

*global fitting — Average diffusion coefficient D [µm²/s]: 77.09

Green - Red Crosscorrelation						
	Napp	G(t,CC)	cRG [nM] (g)	$G_{0,CCF}/G_{0ACFgreen}$	cRG [nM] (r)	$G_{0,CCF}/G_{0ACFred}$	AVG cRG [nM]
2	20.4	0.049	6.30	0.57	9.16	0.72	7.73
3	15.2	0.066	4.57	0.56	6.64	0.71	5.60
4	13.6	0.074	4.26	0.57	4.85	0.64	4.55
5	13.5	0.074	4.02	0.56	5.20	0.67	4.61
6	13.4	0.075	4.13	0.57	5.16	0.67	4.64

$c_{RG} = \frac{G_{0,CCF}}{G_{0,ACF,green}} * c_{green}$ $c_{RG} = \frac{G_{0,CCF}}{G_{0,ACFred}} * c_{red}$

Now we are ready to switch to our real samples measured in live cells.

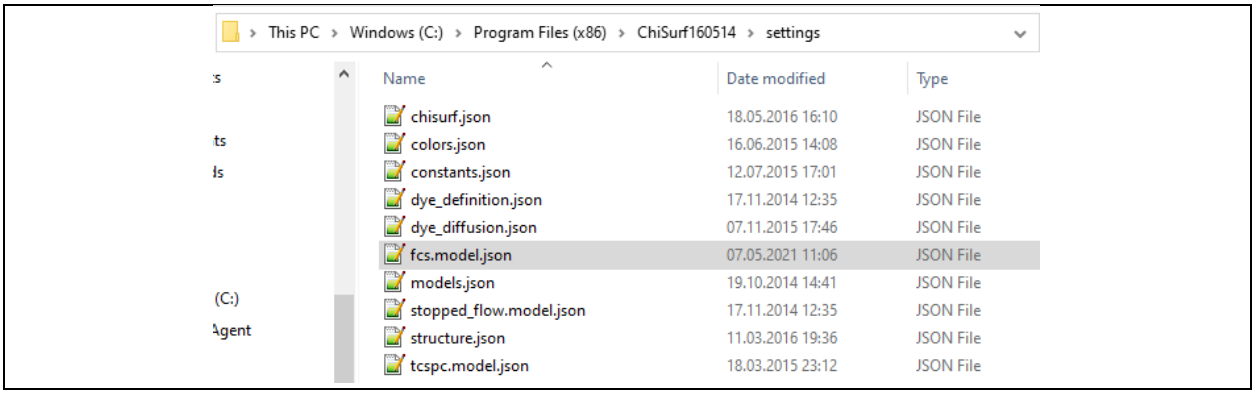
Analysis of live cell experiments

Again, also here, we assume that you have already correlated your data using the software of your system or any of the software mentioned in the beginning.

We will show you first how to add your own fit models to ChiSurf and next analyze first the measurements of the singly labeled constructs (β_2AR-eGFP-IL3 and NT-SNAP-β_2AR) before switching to the double-labeled NT-SNAP-β_2AR-eGFP-IL3 sample.

Adding the membrane-diffusion models

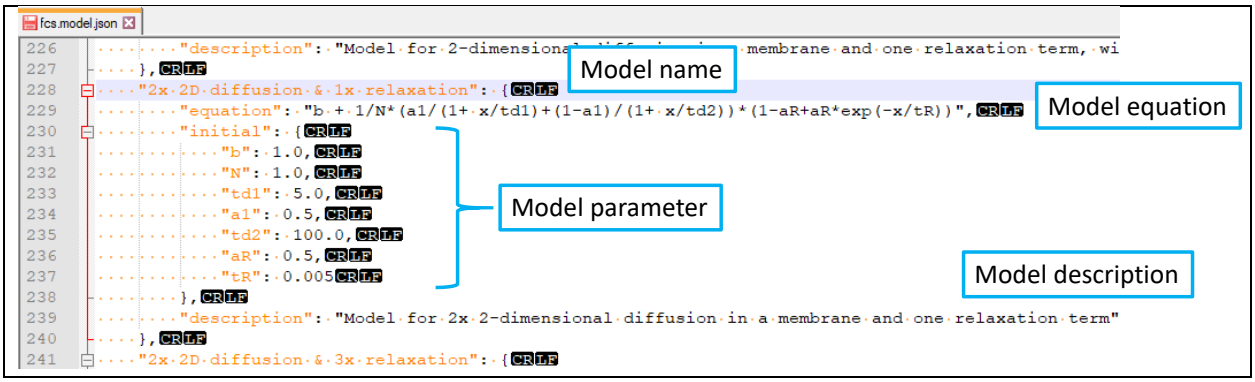
Upon installation, ChiSurf comes with a bunch of FCS fit models; however, your required fit model might not be among them. The fit models are defined in a JSON-file ("fcs.model.json") which can be found in the installation folder of ChiSurf:



Before modifying the file, (i) create a copy on a different place as a backup and (ii) make a second copy to work on as modifying / saving directly in the programs installation folder is usually not allowed.
Open the JSON-file using a text editor, e.g. Notepad++.
Each fit model consists of four sections:

- Model name
- Model equation
- Model parameter definition by initial values
- Model description

It is vital to keep this notation and take care of proper punctuation and indentation!



Add the two following fit models for bimodal membrane diffusion with or without and additional relaxation / triplet term to your JSON-file:

- For analysis of autocorrelation curves:

$$G_{ACF,2D}(t_c) = b + \frac{1}{N}\left[\frac{a_1}{1 + {t_c}/{t_{D1}}} + \frac{1 - a_1}{1 + {t_c}/{t_{D2}}}\right]\left[1 - a_R + a_R \cdot \exp\left(-\frac{t_c}{t_R}\right)\right]$$

where t_{D1} and t_{D2} are the two diffusion time and a_1 is the fraction of t_{D1}. a_R and t_R describe the triplet blinking / photophysics.

- For analysis of cross-correlation curves:

$$G_{CCF,2D}(t_c) = b + \frac{1}{N}\left[\frac{a_1}{1 + {t_c}/{t_{d1}}} + \frac{1 - a_1}{1 + {t_c}/{t_{d2}}}\right]$$

In cross-correlation curves, usually no triplet blinking can be seen.

Don't forget to define reasonable initial values for each of the model parameter.
Replace the original JSON-file in your programs folder with your modified version and restart ChiSurf.
Note: For changes to the JSON-file to take effect, ChiSurf must always be restarted!

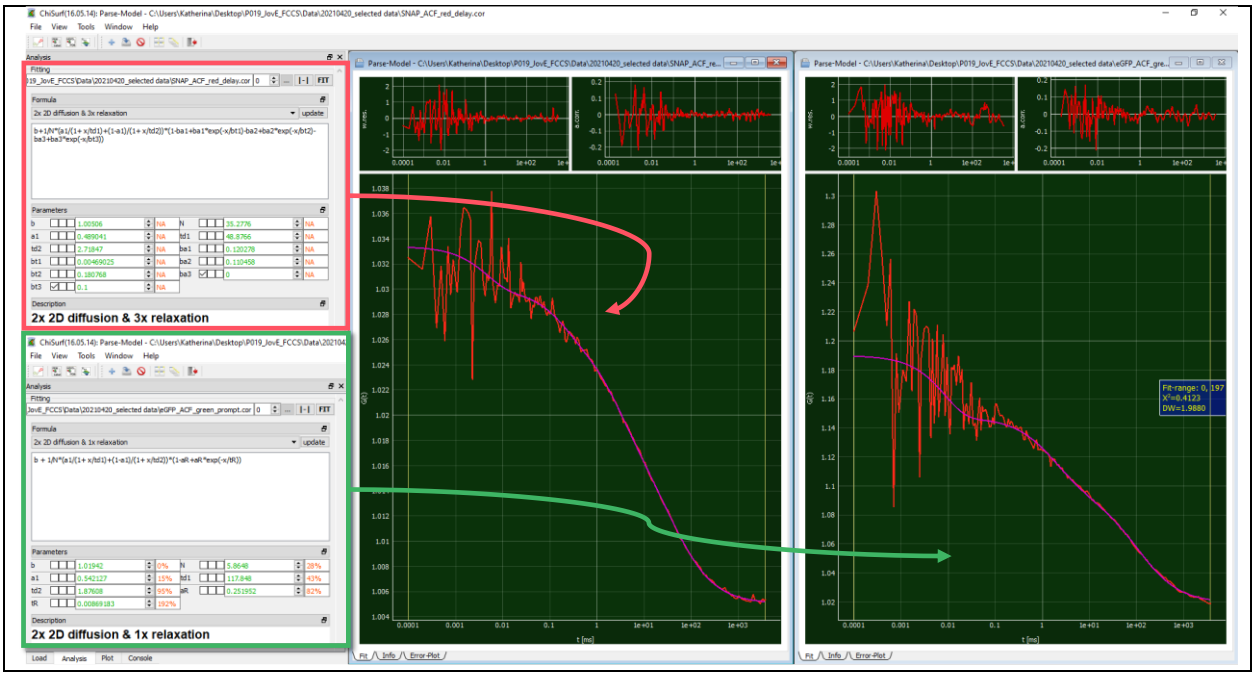
Individual transfected β2AR-eGFP-IL3 and NT-SNAP-β2AR

Fit of autocorrelation curves

Open the autocorrelation of the green channels from the prompt time window from the β2AR-eGFP-IL3 measurements and the autocorrelation of the red channels in the delay time window from the CT-SNAP-β2AR measurements in ChiSurf.
Select the bimodal membrane diffusion model for autocorrelation curves, which we have just added to ChiSurf and fit the data:

For average of the β2AR-eGFP-IL3 measurements, a slow diffusion time t_{D1} = 118 ms and a fraction of 54 % is obtained. The faster diffusion lies at t_{D2} = 1.9 ms. Additionally, 25 % triplet blinking at t_R ~ 9 µs is observed. However, at this short correlation times the data is already quite noisy and care should be taken in the interpretation. The number of molecules in focus is 5.

For the average of the NT-SNAP-β2AR measurements, two additional relaxation terms seem to be required (modify your JSON-model file accordingly!) with relaxation times (and fractions) of t_{R1} ~ 5 µs (12 %) and t_{R2} ~ 180 µs (11%). The two diffusion components show times of t_{D1} = 49 ms (49 %) and t_{D2} = 2.7 ms. The number of molecules in focus lies at 35 and this is much higher compared to β2AR-eGFP-IL3. One could speculate whether t_{R2} ~ 180 µs might not be a photophysics-related term but rather unreacted SNAP substrate diffusing through the confocal volume.



Based on the obtained number of molecules in focus and the known average count rates, we can also determine the molecular brightness of our fluorophores in the live cell settings and estimate the concentration of molecules using the equations explained in the calibration section.
Take care to subtract the background signal e.g. measured on non-transfected cells from the average count rate of your fluorescence samples.

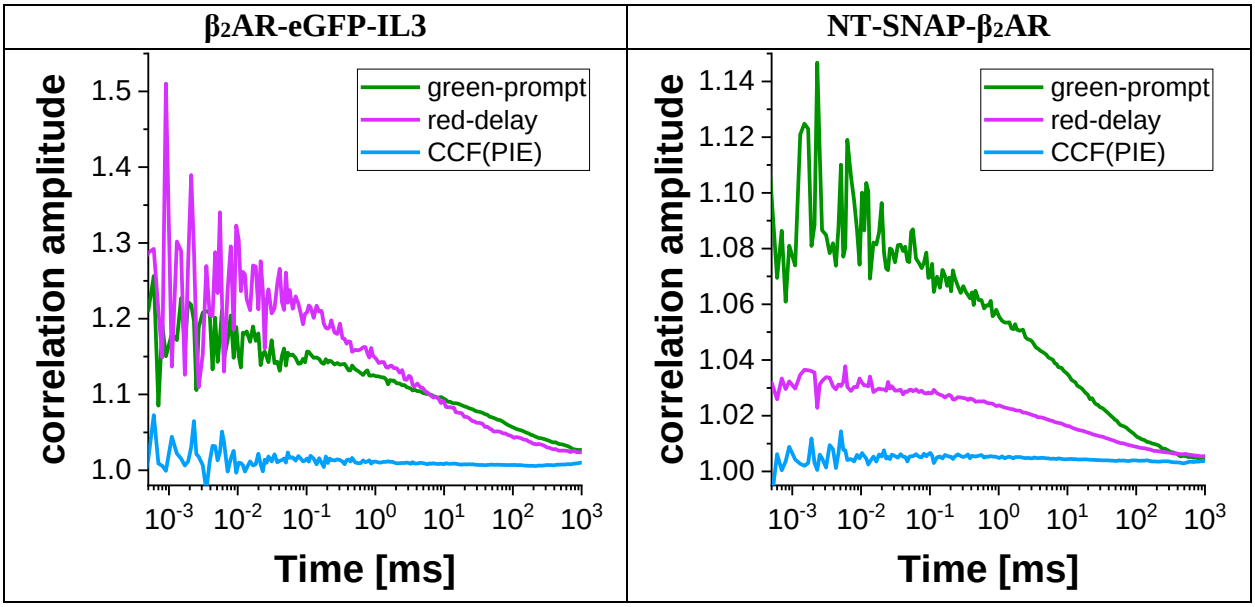
Single-labeled constructs					
Sample	Average countrate [kHz]		Number of molecules	Mol. Brightness [kHz/molecule]	Approx. Concentration [nM]
	s	p			
eGFP	1.87	2.81	5.86	0.80	4.30
SNAP	21.1	39.6	35.28	1.72	67.47

Crosstalk-induced correlations

Based on these single-color experiments also one can test how much artificial / unwanted cross-correlation into the respective other color channel is present.
For this, we export and plot the following correlations:

- From the β_2AR-eGFP-IL3 measurements: red channels in the delay time window and PIE cross-correlation (green channel prompt time window with red channels in the delay time window)
- From the NT-SNAP-β_2AR measurements: green channels in the prompt time window and PIE-cross-correlation (green channel prompt time window with red channels in the delay time window)

In ideal case, all of these combinations show flat curves or better-said noise distributed around 1. Here, this is the case only for the PIE-cross-correlations. The respective "false-color" autocorrelation functions, red-delay from β_2AR-eGFP-IL3 and green-prompt from CT-SNAP-β_2AR, reflect the crosstalk into the red channels and the direct excitation of red fluorophore by the green excitation wavelength, respectively.

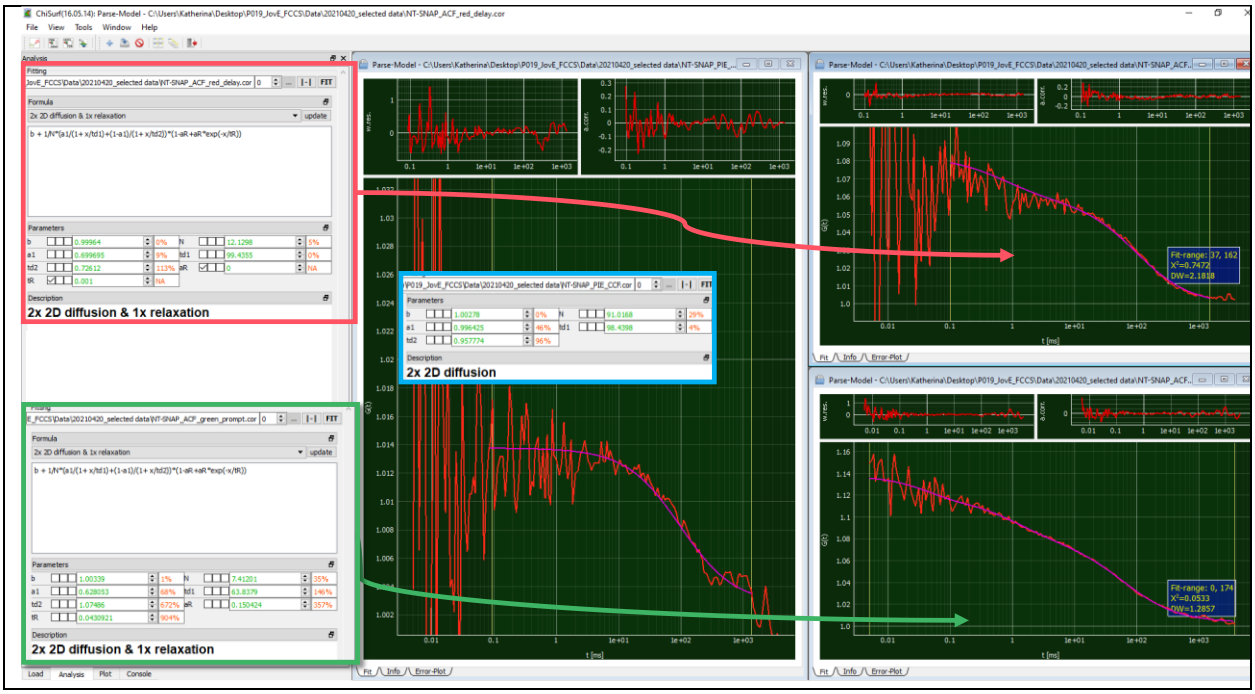


Double-labeled sample: NT-SNAP-β2AR-eGFP-IL3

Fit of auto- and cross-correlation curves

Open all three average curves, the green-prompt autocorrelation, red-delay autocorrelation and PIE-cross-correlation curve in ChiSurf. For both autocorrelation curves, we add the fit model with bimodal membrane diffusion and an additional relaxation term. For the PIE-cross-correlation curve, a bimodal membrane diffusion model is sufficient.

For the red autocorrelation and especially the PIE-cross-correlation, the fit ranges have to be adjusted as particularly for short correlation curves the noise is very high and no reliable fit could be acquired here.



Calculation of co-diffusing molecules

Based on the (apparent) number of molecules in focus and our determined correction factors for the confocal overlap volume from the DNA measurements, the fraction of double-labeled molecules can be calculated.

Double-labeled construct: NT-SNAP & IL3-eGFP

Sample	Average countrate [kHz]		App. number of molecules	Approx. Concentration [nM]	ratio G_0(tc)	fraction co-diffusion
	s	p				
Green-prompt	3.16	4.85	7.41	3.56		
Red-delay	0.53	1.45	12.13	2.78		
PIE			91.0	0.52	0.08	0.15
				0.72	0.13	0.26

Example calculation:						
DNA -> 100 % co-diffusion						
$G_{0ACF,green}$ =	1		$r_{green,ideal}$=	0.5		
$G_{0ACF,red}$ =	0.8		$r_{red,ideal}$=	0.625		
G_{0CCF} =	0.5					
Cell sample shows					fraction co-diffusion:	
$G_{0ACF,green}$ =	0.12		$r_{green,cell}$=	0.167	33.3	%
$G_{0ACF,red}$ =	0.08		$r_{red,cell}$=	0.25	40.0	%
G_{0CCF} =	0.02					

Here, N_{eGFP} = 7.4, N_{SNAP} = 12.1 and $N_{app,PIE}$ = 91. Thus, the amplitudes are zero correlation time G(t_c=0) have the following values: $G_{eGFP}(0)$ = 0.135, $G_{SNAP}(0)$ = 0.082, and $G_{PIE}(0)$ = 0.011.
Using the correction factors from the DNA measurement, here only between 15 -26 % molecules show both labels. However, (i) the data is quite noisy and (ii) the correlation amplitudes are very low. Both factors lead to large errors.
Conclusion: Search for cells with low expression level, i.e. low fluorescence and take your time to collect a decent amount of photons to correlate!

Analysis of simulated data

Simulation details

The simulations of the β_2AR-eGFP-IL3-CT-SNAP measurements (short: CTSNAP) were performed using Burbulator [5] (part of the MFD software package, https://www.mpc.hhu.de/software/3-software-package-for-mfd-fcs-and-mfis).
The NTSNAP construct has both fluorophore on the inner side in a membrane and we assume (i) the fluorophore to be close enough to each other to undergo FRET and (ii) the membrane receptor β_2AR to show dynamics such that the fluorophores exchange between two different levels of FRET.
Burbulator uses the Becker&Hickl spc-file format with 4096 TAC channel with a width of 4.07 ps and a laser period of 13.596 ns. The green-to-red detection efficiency ratio and the g-factor was set 1 and the fundamental anisotropy to 0.38. The fluorescence lifetime of both eGFP and SNAP was set to 3 ns with a molecular brightness of 10 kHz/molecule, fluorescence quantum yield of 0.8 and a rotational correlation time of 100 ns. Additionally the background in the green channel was set to 1 kHz and in the red channel to 0.5 kHz. The green crosstalk into the red channels was set to 0.1. All listed values for the fluorophores were adopted based on our measurements from the NTSNAP construct and the setup-describing values were set to reasonable values or 1.
The mean lifetime of the low FRET (LF) and high FRET (HF) states for dynamic exchange was set to 2.4 ns (E = 0.2) and 0.9 ns (E = 0.7). The equilibrium fractions of LF and HF were set to 0.5 each with a relaxation rate of 71 µs and – in case of triplet – with 16 % triplet blinking at 5.5 µs (Caution: Burbulator adds triplet blinking only to donor molecules!).
The diffusion term was modeled as a bimodal distribution with 30 % of fast diffusing molecules at t_{D1} = 1 ms and the rest of the molecules diffusing slowly with t_{D2} = 100 ms.
In total, 10^7 photons were simulated in a 3D Gaussian shaped volume with w_0 = 0.5 µm and z_0 = 1.5 µm, a box size of 20, and N_{FCS} = 0.01.
Of note: As the slower modeled diffusion time is quite long, the number of photons and the box size might have to be increased further to allow good fitting at large correlation times t_c. Here, for the sake of time / simplicity, the fitting was stopped at t_c = 1 sec.

Global fit of auto- and cross-correlation curves

Before starting ChiSurf, add the fit model for the cross-correlation curve to your JSON-file as described above:

$$G_{CCF,2D}(t_c) = b + \frac{1}{N}\left[\frac{a_1}{1+{}^{t_c}/_{t_{d1}}} + \frac{1-a_1}{1+{}^{t_c}/_{t_{d2}}}\right]\left[1 - a_R \cdot \exp\left(-\frac{t_c}{t_R}\right)\right]$$

where a_R and t_R describe the amplitude and relaxation time of the anticorrelation.

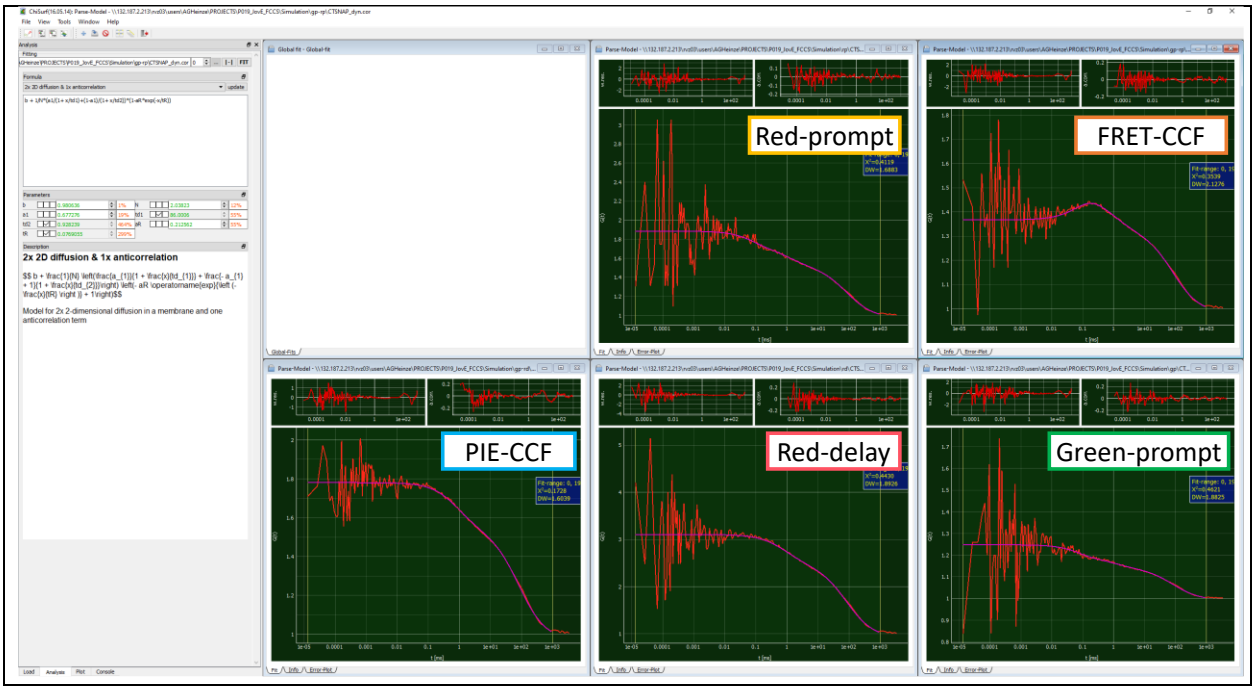
A more general equation – in case of more than one relaxation term – would have the following form:

$$G_{CCF,2D}(t_c) = b + \frac{1}{N}\left[\frac{a_1}{1+{}^{t_c}/_{t_{d1}}} + \frac{1-a_1}{1+{}^{t_c}/_{t_{d2}}}\right]\left[1 - a_f\right]\left[1 - \sum_i a_{Ri} \cdot \exp\left(-\frac{t_c}{t_{Ri}}\right)\right]$$

where a_f describes the total amplitude of the anticorrelation (identical to a_R in the single anticorrelation term model above) and a_{Ri} and t_{Ri} the respective relaxation times and amplitudes.
Load in total five different correlation curves into ChiSurf:

- Green-prompt (autocorrelation of green signal in prompt time window)
- Red-prompt (autocorrelation of the FRET-induced red signal in the prompt time window)
- Red-delay (autocorrelation of the red signal (direct excitation) in the delay time window)
- FRET-CCF (cross-correlation of green prompt and red-prompt signal)
- PIE-CCF (cross-correlation of green-prompt with red-delay)

The two new curves (red-delay and FRET-CCF), which we have not used to far, both stem from the FRET-induced red signal now present in our data.



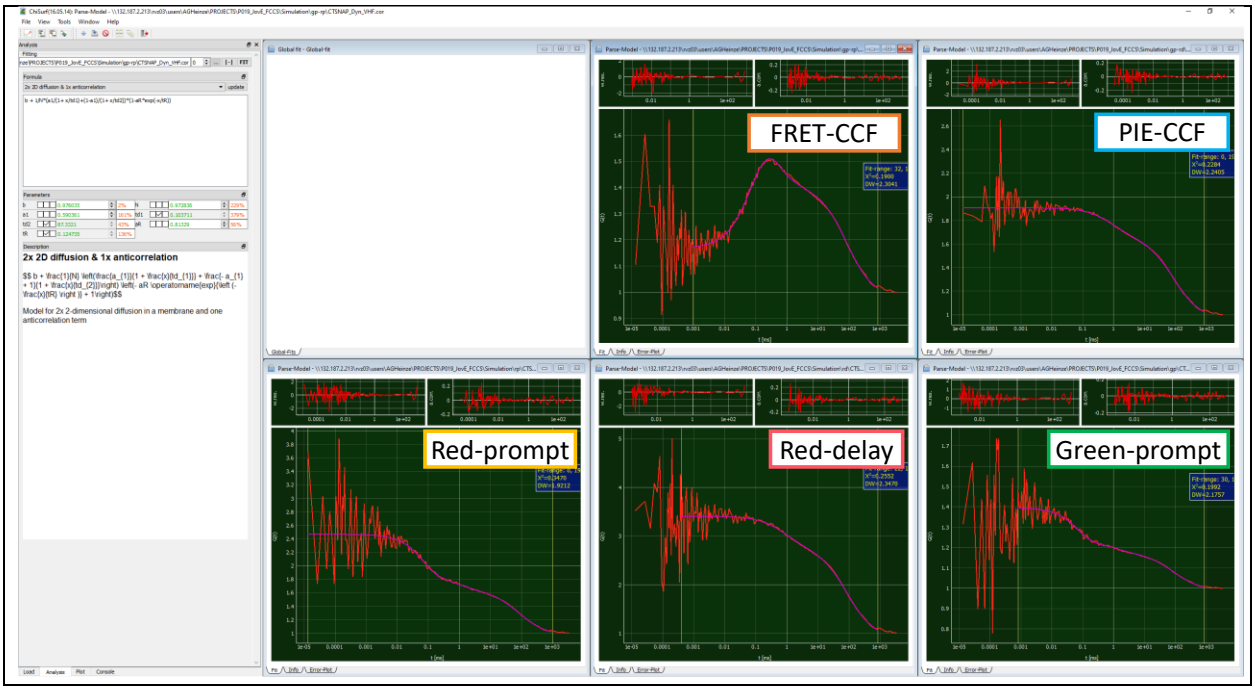
Due to the FRET-induced anticorrelated behavior of green and red signal in the prompt time window. The FRET-CCF shows a “dip” at short correlation time, coinciding with a rise in both autocorrelation curves from the prompt time window.

All five loaded curves are fit jointly with linked t_{D1}, t_{D2} and t_R. The fit results are summarized in the table below. Please note that here the diffusion times can be fit jointly as the simulation software does not support the modelling of differently sized confocal detection volumes. For experimental results, this joint fitting of t_D might not be possible, however t_R should be linked.

Parameter	green-prompt	red-prompt	red-delay	FRET-CCF	PIE-CCF
N(app)	3.90	1.09	0.455	2.04	1.23
t_{D1} **[ms]**			86.0		
a_1	0.710	0.687	0.694	0.677	0.687
t_{D2} **[ms]**			0.928		
a_R	0.203	0.144	0	0.213	0
t_R **[µs]**			76.9		

Influence of FRET efficiency

The extent of the anticorrelation can be used also as a marker for the extent of change in FRET efficiency E: If we change the LF state to $E = 0$ and the HF state to $E = 0.95$, the induced dip in the FRET-CCF is much more pronounced than for the first example.



The amplitude of the anticorrelation has increased to a_R = 0.66 compared to an a_R = 0.21 from the previous case.

Parameter	green-prompt	red-prompt	red-delay	FRET-CCF	PIE-CCF
N(app)	2.45	0.664	0.397	0.973	1.05
t_{D1} **[ms]**			91.8		
a_1	0.722	0.704	0.728	0.713	0.77
t_{D2} **[ms]**			0.886		
a_R	0.39	0.392	0	0.662	0
t_R **[µs]**			70.5		

Influence of triplet blinking

Be aware that significant amount of triplet blinking in the µs time range of the donor and / or acceptor fluorophore might mask the anticorrelation induced due to FRET in the FRET-CCF.
In the example shown below, 16 % of additional triplet blinking at 5.5 µs was added to the example of LF(E = 0.2) <-> HF(E = 0.7).

Here, the FRET-CCF model has to be extended for a triplet-induced correlation term:

$$G_{CCF,2D}(t_c) = b + \frac{1}{N}\left[\frac{a_1}{1+{t_c}/{t_{d1}}} + \frac{1-a_1}{1+{t_c}/{t_{d2}}}\right]\left[1 - a_R \cdot \exp\left(-\frac{t_c}{t_R}\right)\right]\left[1 - a_T + a_T \cdot \exp\left(-\frac{t_c}{t_T}\right)\right]$$

where a_T and t_T describe the amplitude and relaxation time of the triplet component.

One can see quite nicely in the FRET-CCF how the "dip" in the curve due to the anticorrelation term is counteracted by the triplet component.

Parameter	green-prompt	red-prompt	red-delay	FRET-CCF	PIE-CCF
N(app)	4.79	1.73	0.611	1.73	1.57
t_{D1} **[ms]**			98.0		
a_1	0.678	0.670	0.677	0.670	0.674
t_{D2} **[ms]**			1.12		
a_R	0.210	0.115	0	0.115	0
t_R **[µs]**			35.3		
a_T	0.146	0.015	0.113	0.015	0.171
t_T **[µs]**			7.89		

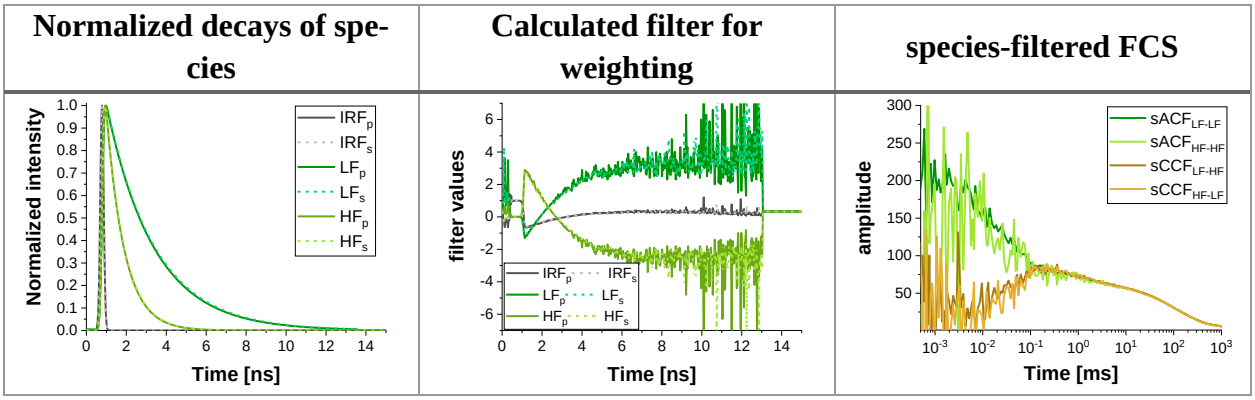
Species-filtered FCS to recover dynamics

A method to recover the triplet-masked anticorrelations in the FRET-CCF is to make use of the microtimes (i.e. the fluorescence decay histograms) encoded in the data. Here, instead of direct photon traces, an additional weighting function is introduced based on the fluorescence decay shape of the (i) IRF, (ii) the LF state, and (iii) the HF state. More details on how to generate these weighting functions can be found e.g. in the following literature [6,7].
In this species-specific or filtered FCS approach, four different correlation pattern are generated:

- Species-autocorrelation of the LF state ($sACF_{LF\text{-}LF}$)
- Species-autocorrelation of the HF state ($sACF_{HF\text{-}HF}$)

- Species-cross-correlation of the LF state to the HF state ($sCCF_{LF\text{-}HF}$)
- Species-cross-correlation of the HF state to the LF state ($sCCF_{HF\text{-}LF}$)

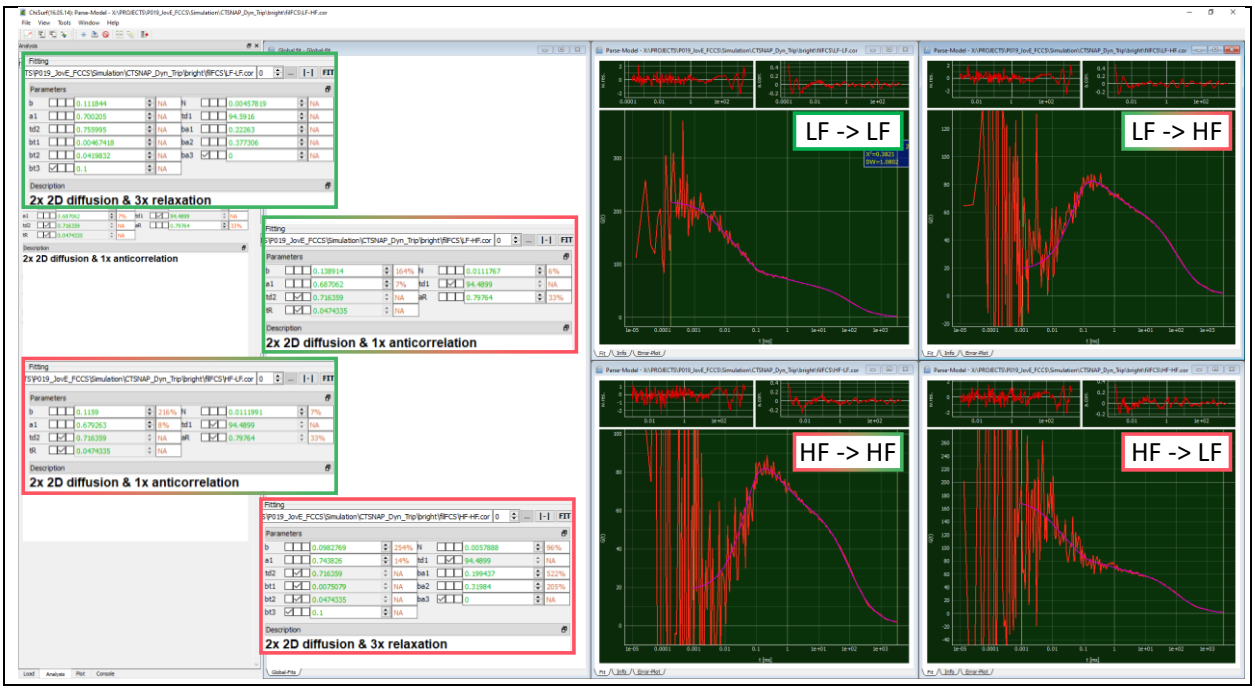
Below, exemplary the work flow and input for a filteredFCS analysis of the LF(E = 0.2) <-> HF(E = 0.7) example with additional triplet is shown.



Note: Suffix "p" and "s" are used to discriminate between the parallel (p) and perpendicular (s) channel here.

For generation of the species-filtered FCS curves, the weights determined for the LF and HF species based on the normalized intensity decays are used during the correlation.

The resulting curves are fit to standard equations with bimodal membrane diffusion and relaxation and anticorrelation terms, respectively.



During the fit, both diffusion times t_{D1} and t_{D2} as well as the relaxation times are fit jointly and the FRET-induced relaxation of the $sACF_{LF\text{-}LF}$ and $sACF_{HF\text{-}HF}$ is linked to the anticorrelation term of the $sCCF_{LF\text{-}HF}$ and $sCCF_{HF\text{-}LF}$.
Be aware that the number of molecules in focus, N, is only an apparent number and does no longer relate to the concentration of molecules in the experiment / simulations!

Parameter	LF->LF	HF->HF	LF->HF	HF->LF
N(app)	0.0046	0.0058	0.011	0.011
t_{D1} **[ms]**	94.6			
a_1	0.700	0.687	0.687	0.679
t_{D2} **[ms]**	0.756			
a_R	0.378	0.320	0.798	
t_R **[µs]**	42.0			
a_T	0.223	0.199	0	0
t_T **[µs]**	4.67			

References

1 Peulen, T. O., Opanasyuk, O. & Seidel, C. A. M. Combining Graphical and Analytical Methods with Molecular Simulations To Analyze Time-Resolved FRET Measurements of Labeled Macromolecules Accurately. *J Phys Chem B.* **121** (35), 8211-8241, (2017).

2 Petrasek, Z. & Schwille, P. Precise measurement of diffusion coefficients using scanning fluorescence correlation spectroscopy. *Biophys J.* **94** (4), 1437-1448, (2008).

3 Siegel, A. P., Baird, M. A., Davidson, M. W. & Day, R. N. Strengths and weaknesses of recently engineered red fluorescent proteins evaluated in live cells using fluorescence correlation spectroscopy. *Int J Mol Sci.* **14** (10), 20340-20358, (2013).

4 Hellenkamp, B. *et al.* Precision and accuracy of single-molecule FRET measurements-a multi-laboratory benchmark study. *Nat Methods.* **15** (9), 669-676, (2018).

5 Kalinin, S., Valeri, A., Antonik, M., Felekyan, S. & Seidel, C. A. Detection of structural dynamics by FRET: a photon distribution and fluorescence lifetime analysis of systems with multiple states. *J Phys Chem B.* **114** (23), 7983-7995, (2010).

6 Bohmer, M., Wahl, M., Rahn, H. J., Erdmann, R. & Enderlein, J. Time-resolved fluorescence correlation spectroscopy. *Chemical Physics Letters.* **353** (5-6), 439-445, (2002).

7 Felekyan, S., Kalinin, S., Sanabria, H., Valeri, A. & Seidel, C. A. M. Filtered FCS: Species Auto- and Cross-Correlation Functions Highlight Binding and Dynamics in Biomolecules. *Chemphyschem.* **13** (4), 1036-1053, (2012).

7.2. Determination of g-factor and depolarization factors using ChiSurf

High NA objectives cause a partial depolarization of the linearly polarized excitation and the collected emitted light. This influences the observed fluorescence anisotropy in fluorescence spectroscopy experiments:

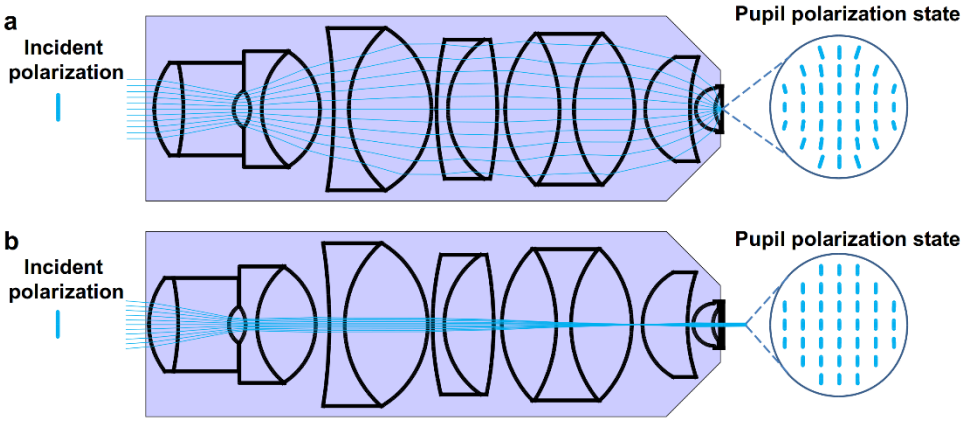


Figure 1. Depolarization introduced by a high NA microscope objective.

(a) The microscope objective is illuminated by a linearly polarized plane wave and focuses the beam onto the sample. This mode of illumination is used in confocal microscopy. (b) The linearly polarized beam is focused into the back-focal plane of the objective (widefield illumination). The sample is illuminated by a collimated plane wave. The "pupil" polarization states in the image planes are depicted using vertically polarized incident beams. Widefield illumination (b) leads to lower loss of polarization in the illumination field than focused beam illumination (a) as confirmed by optical ray tracing simulations.

(Image taken from: https://doi.org/10.1371/journal.pone.0100526)

The correction of this phenomenon has first been described already by M. Koshioka, K. Sasaki and H. Masuhara (https://opg.optica.org/as/abstract.cfm?uri=as-49-2-224, , Appl. Spectrosc., 1995, 49, 224–228).

In the more recent, open access publication of Erdelyi *et al.* a more illustrative description can be found (Erdelyi M, Simon J, Barnard EA, Kaminski CF (2014) Analyzing Receptor Assemblies in the Cell Membrane Using Fluorescence Anisotropy Imaging with TIRF Microscopy. PLOS ONE 9(6): e100526. https://doi.org/10.1371/journal.pone.0100526).

Next to the depolarization due to the objective, the different detection sensitivity of the parallel and perpendicular detector must be considered. In microscopy-based experiments, this so-called g-factor is often defined as the ratio of parallel (I_p) over perpendicular (I_s) light, however, some software also uses the inverse definition. In either case, the g-factor should ideally lie close to 1.

To determine both the g-factor and polarization correction factors, l_p, and l_s, we use here a joint analysis of a small, fast rotating fluorophore such as Alexa, Atto or Cy fluorophores and a larger, slow rotating fluorophore such as a fluorescent protein. Please note that both selected fluorophores must show a mono-exponential rotation, i.e. proteins labelled with an organic fluorophore connected via a flexible maleimide linker or similar are not recommended.

Joint analysis with ChiSurf

Data input and file format

ChiSurf can read a variety of text files and formats. For time-resolved fluorescence intensities in polarization-resolved experiments, we usually use two different file formats: (ii) two-columns/two files or (ii) single-column/single file ("Jordi-format").

In the first case, two files are required, one containing the parallel channel and one the perpendicular channel data. Both files contain two columns; the first column is the time in nanoseconds, while the second column contains the actual data (photon counts in this respective time bin).

In the second case, only a single file is required, in which the data from perpendicular and parallel channel are stacked on top of each other.

Please note that in both format no header is used, however, it can be defined to skip header rows while file loading.

Case I: two files / two columns	Casse II: single file ("Jordi" format)
eGFP_44_ROI1_green_s.txt | eGFP_44_ROI1_green_p.txt | A488_green_jordi.txt 1 0.00000000000000000e+00 2.40000000000000000e+01 2 2.00000000000000042e-02 2.40000000000000000e+01 3 4.00000000000000083e-02 3.30000000000000000e+01 4 5.99999999999999778e-02 3.00000000000000000e+01 5 8.00000000000000167e-02 2.20000000000000000e+01 6 1.00000000000000056e-01 3.20000000000000000e+01 7 1.19999999999999956e-01 2.70000000000000000e+01 8 1.40000000000000133e-01 3.00000000000000000e+01 9 1.60000000000000033e-01 3.20000000000000000e+01 10 1.79999999999999933e-01 4.10000000000000000e+01 11 2.00000000000000111e-01 3.30000000000000000e+01	A488_green_jordi.txt 1 3.20000000000000000e+01 2 4.60000000000000000e+01 3 5.20000000000000000e+01 4 4.20000000000000000e+01 5 3.80000000000000000e+01 6 4.70000000000000000e+01 7 5.10000000000000000e+01 8 4.30000000000000000e+01
➔ File 1: 2500 lines in total, parallel	➔ Lines 1- 2500 = parallel

eGFP_44_ROI1_green_s.txt | eGFP_44_ROI1_green_p.txt | A488_green_jordi.txt

```
1   0.00000000000000000e+00 2.70000000000000000e+01
2   2.00000000000000042e-02 1.60000000000000000e+01
3   4.00000000000000083e-02 2.00000000000000000e+01
4   5.99999999999999778e-02 1.20000000000000000e+01
5   8.00000000000000167e-02 2.80000000000000000e+01
6   1.00000000000000056e-01 2.00000000000000000e+01
7   1.19999999999999956e-01 1.90000000000000000e+01
8   1.40000000000000133e-01 1.50000000000000000e+01
9   1.60000000000000033e-01 1.60000000000000000e+01
10  1.79999999999999933e-01 1.60000000000000000e+01
```

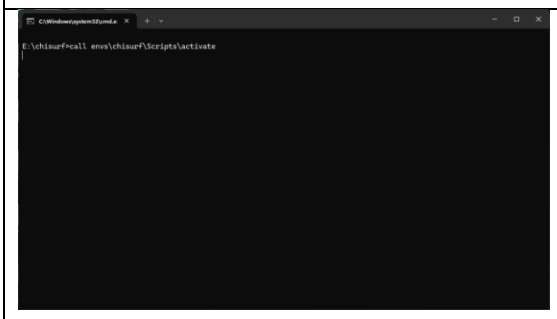
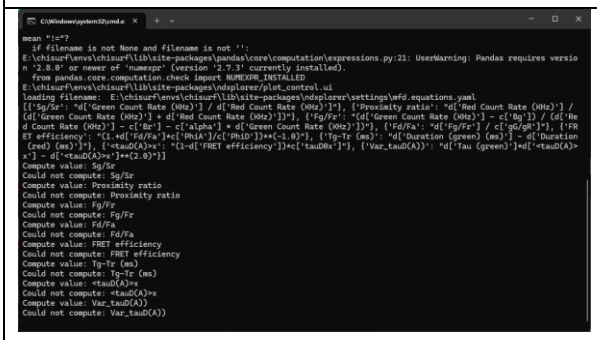
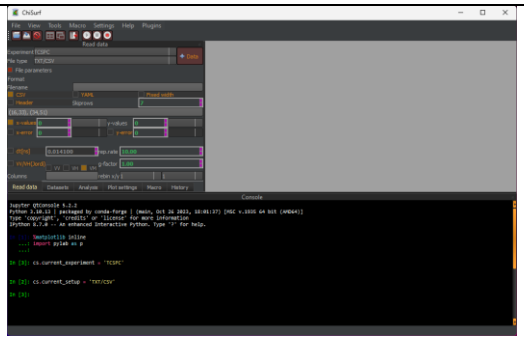
➔ File 2: 2500 lines in total, perpendicular

A488_green_jordi.txt

```
2497   0.00000000000000000e+00
2498   0.00000000000000000e+00
2499   0.00000000000000000e+00
2500   0.00000000000000000e+00
2501   7.20000000000000000e+01
2502   1.11000000000000000e+02
2503   9.00000000000000000e+01
2504   8.60000000000000000e+01
2505   1.02000000000000000e+02
```

➔ Lines 2501 – 5000 = perpendicular

I personally prefer the single-column Jordi-format as it reduces the amount of files in my data export folders by 50%, however, I need of course to remember with which time resolution (here 20 ps) the TCSPC histograms were exported.

1:command window:	2. ChiSurf startup window
	
3. command window (final)	4. ChiSurf Main window
	

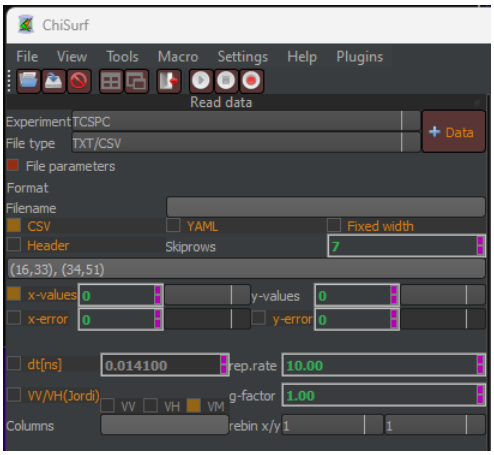
Loading the data: Defining the reading parameter

As ChiSurf is been developed to support a variety of experiment types (TCSPC, FCS, Modelling) and file formats (e.g. txt or csv, tttr, sdt,...) it is vital to define the correct reading parameter.

Here, our *experiment* is “TCSPC” and the *File Type* is “TXT/CSV”.

Two single files

If the parallel and perpendicular data is saved in two separate files, including a time axis, the following reading parameter should be set:

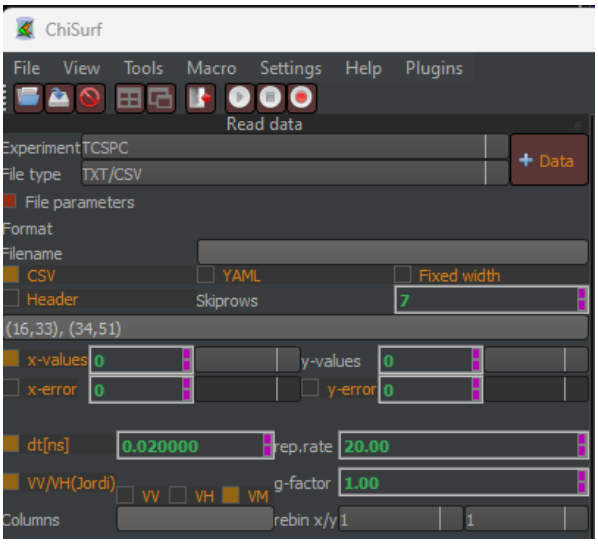


Note that if the files would contain a header, the checkbox next to *Header* must be selected and the number of rows to skip must be indicated in the *Skiprows* box.

Both *g-factor* and repetition rate (*rep.rate*) can already be fixed here. Giving the correct repetition rate is required in single-molecule experiments where the high repetition rates do not allow a full intensity decay of the fluorophores, and the repeated excitation must be considered in the fluorescence lifetime analysis.

Jordi format

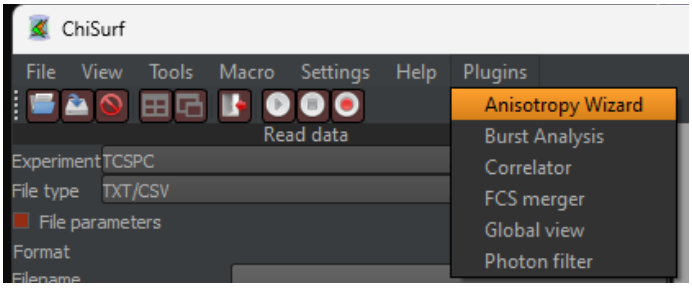
If both parallel and perpendicular data is saved within the same file and no time axis is defined, the following settings need to be made:



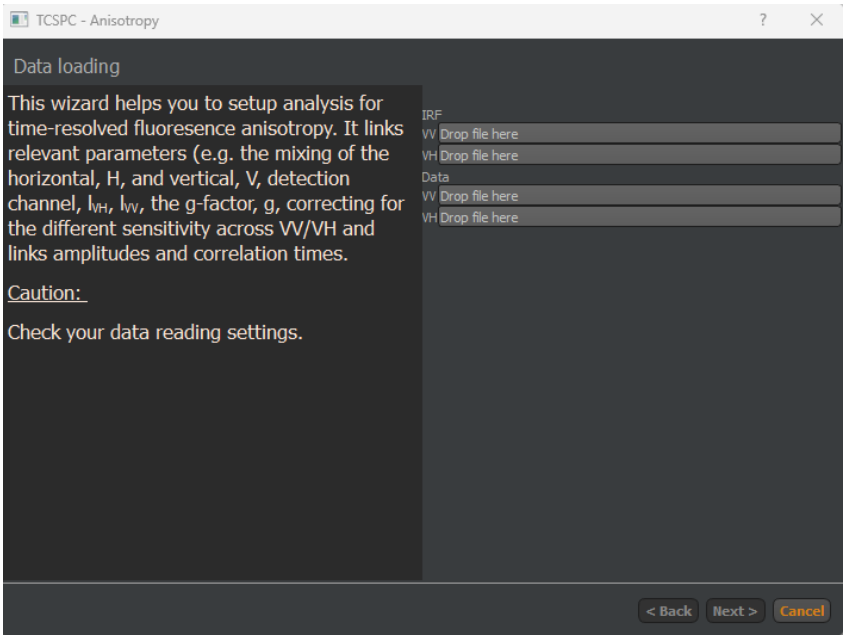
The example data was collected in total 40 MHz PIE mode, i.e. every fluorophore got excited with at 20 MHz, and the data was exported with a time resolution of 20 ps.

Loading the Alexa488 data into the anisotropy plugin

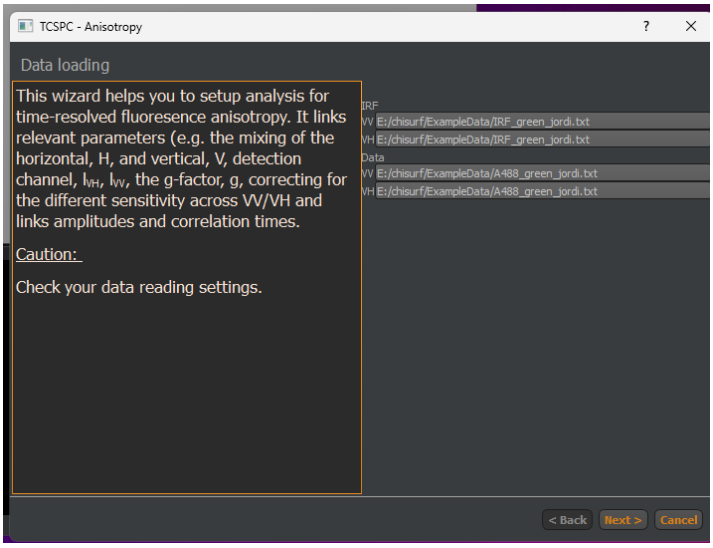
After the reading parameter have been correctly set, the anisotropy wizard can be started from the "Plugins" tab:



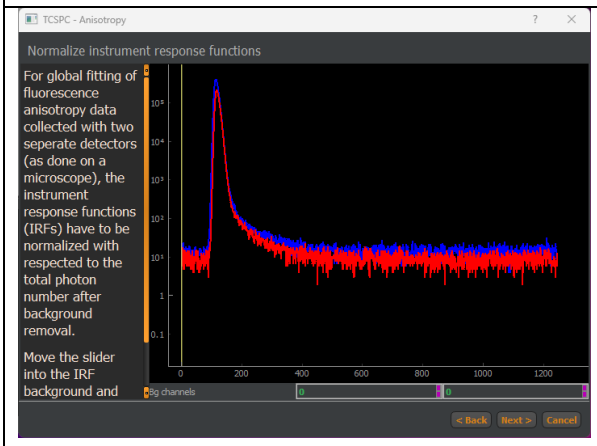
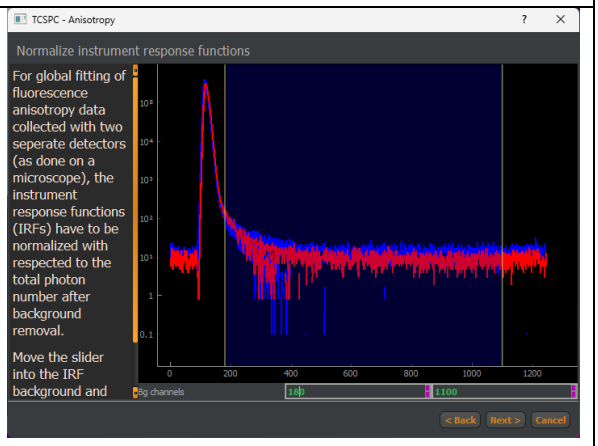
The anisotropy wizard opens in a new window and leads step-by-step through the joint fit setup.



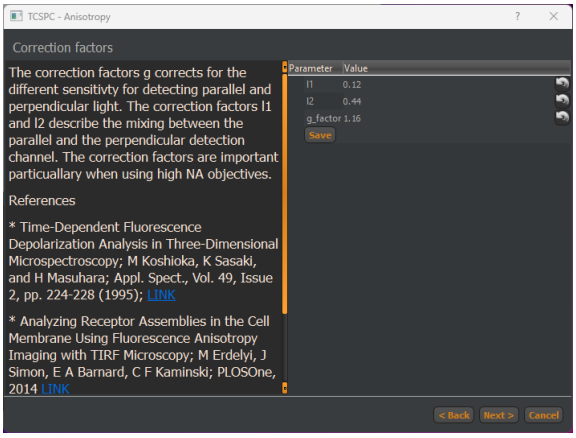
Load the data by "drag'n'drop" the files into the respective boxes. In case the single-file format is used, the still the same file must be loaded twice (The software will automatically load either the parallel or perpendicular section of the data column).



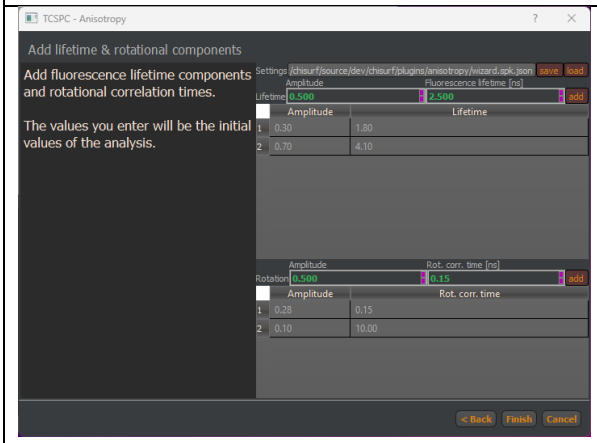
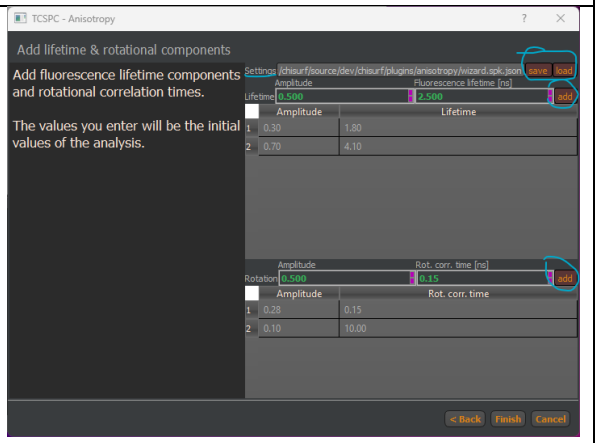
Click “next” to proceed. IN the next step, the IRF background region is defined. Note: The number in the 2nd spin box must always be largr than in the 1st box, i.e. best is to first define the end (here: 100) and then the beginning (here:180)

Define background region of IRF - empty	With correct numbers
	

Cleck “next” to proceed. Here, the *g-factor*, *l1* and *l2* can be given (in case known) or estimates can be added.



Click “next” to continue. IN this step, the fluorescence lifetimes and rotational correlation times can be added. In this step, settings can also be saved and loaded, e.g. if many similar cells need to be analysed (see also below).

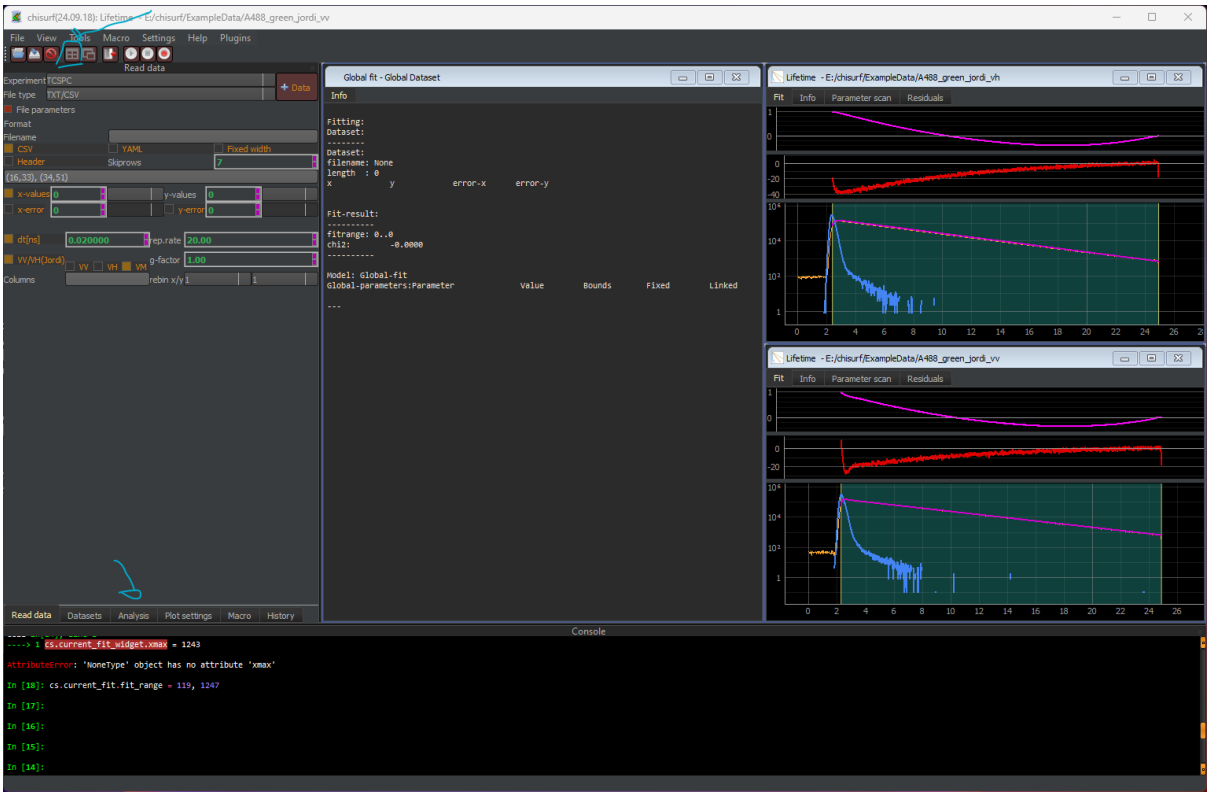
Add LT & rotation - defaults	For Alexa488
	

To add a fluorescence lifetime or a rotational component, adjust the values in the box below “Amplitude”, “Fluorescence Lifetime” and “Rot. Corr. Time” and press “add”.

Components can be removed by double-clicking on the entry in the table.

Please note that the amplitude sum of the fluorescence lifetime components is automatically normalized to 1, while for the anisotropy components the sum is normalized to 0.38, the fundamental anisotropy of most fluorophores.

Click “finish” to generate the joint fits in the main ChiSurf window and press the “tile windows” to distribute the windows in ChiSurf’s main window.



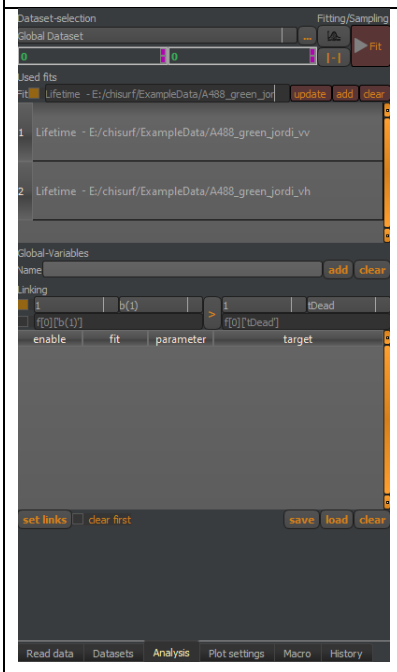
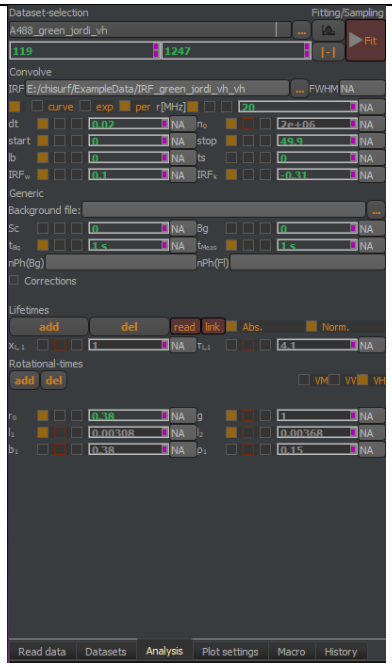
In total, three fit windows are now open. Here, the large window entitled “Global fit” hosts the connections/linking between the two individual fits open in the two smaller windows (here: top contains the perpendicular data, while bottom contains the parallel data).

Next, switch to the “analysis” ta and inspect what is displayed for the three different windows:

- Global-fit: In the top-part the fits, which are to be minimized jointly are listed, the bottom part could be used to manually link variables across different datasets listed above (not used here).
- Lifetime - _vh and Lifetime - _vv: Each of them have four different sections:
    - Convolve (top left to bottom right):
        - Datapath to the IRF
        - Convolution algorithm: e.g. exponential or periodic (selected here)
        - dt: size of a time bin in nanosecond
        - n0: total number of photons in the decay
        - start / stop: start and stop of time range
        - lb: offset of the IRF, must be zero or another small number as we background was already subtracted when setting up the experiment
        - ts: time shift between IRF and decay
        - IRF_w/IRF_k: In case an IRF cannot be measured/is missing, these parameters can be used to generate a synthetic Gaussian-shaped IRF with width IRF_w

and skewness IRF_k. This option is not when (i) an IRF is loaded and (ii) the parameter are fixed (1st of the three boxes selected)

- Generic:
  - Sc: fraction of scatter-based fluorescence, should be low in all experiments except signle-molecule
  - Bg: background/offset of the data
  - tBG/tMeas: to be filled when fitting single-molecule in "Burst-Integrated Fluorescence Lifetime" mode
  - Corrections: Ticking this box reveals option to perform a deadtime correction (pulse pile-up at high count rates) or to correct for differential non-linearities of the counting electronics (white light reference measurement required)
- Lifetime:
  - xL,[x] are the amplitudes of the fluorescence lifetimes, normalized to a sum of 1 if the "Norm." box is ticked. Amplitudes can also get negative (e.g. for FRET-sensitized acceptor emission data), then the box next to "Abs." must be unticked.
  - τL,[x] are the respective fluorescence lifetime.
- Rotational times
  - VM - VV - VH: designate the polarization of the dataset
  - r0: fundamental anisotropy of the used fluorophore
  - g: g-factor
  - l1, l2: polarization correction factors
  - b[x]: amplitudes of the rotational correlation times, normalized to r0

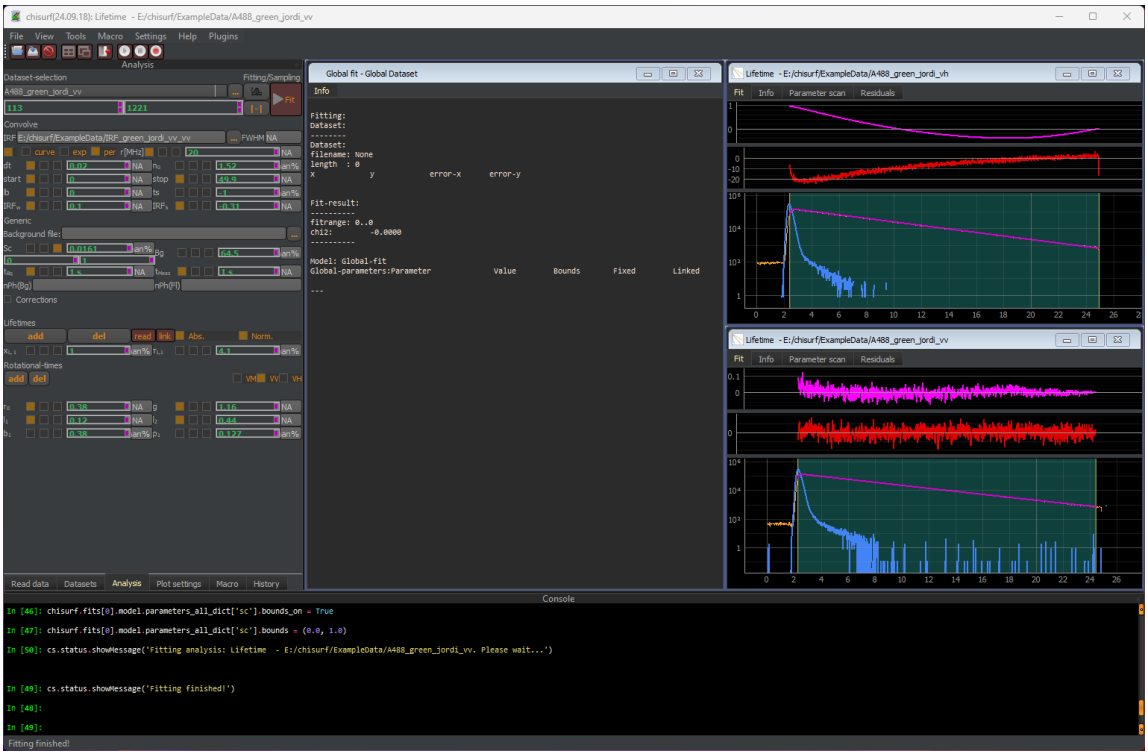
Global fit	Vh	vv
		

Each of the number-based parameter has three checkboxes:

- The left checkbox fixes the parameter, it will not be fit but kept constant.
- The middle checkbox is used to link parameter (via right-click -> link) and if it is filled or shown in read, this means that this parameter has been linked to another data set.
- The right checkbox can be used to define a range, in which the parameter can float, e.g. a fluorescence lifetime should not become negative.

Step 1: Approximating g-factor with small fluorophore

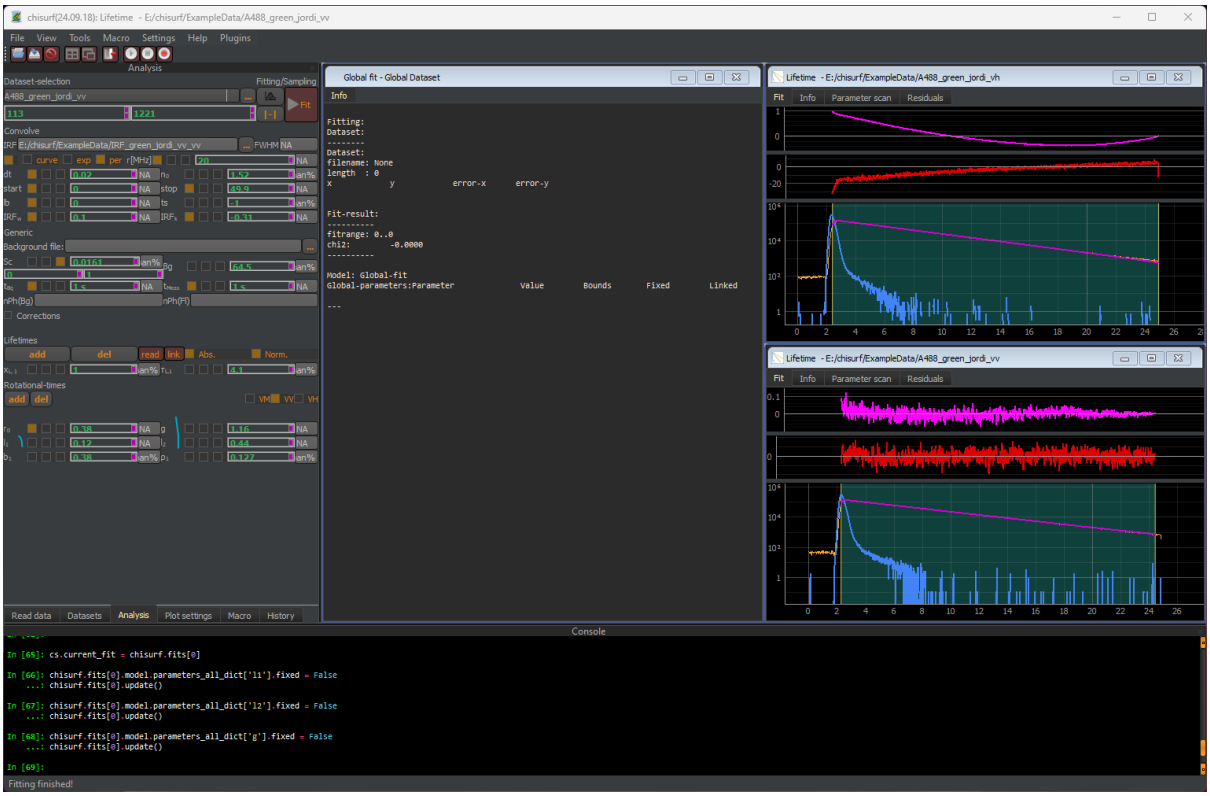
To start the analysis, first go to the “VV” dataset and perform a quick, individual fit of this dataset. This helps to set at least the time shifts between IRF and decay of the background/offset correctly.



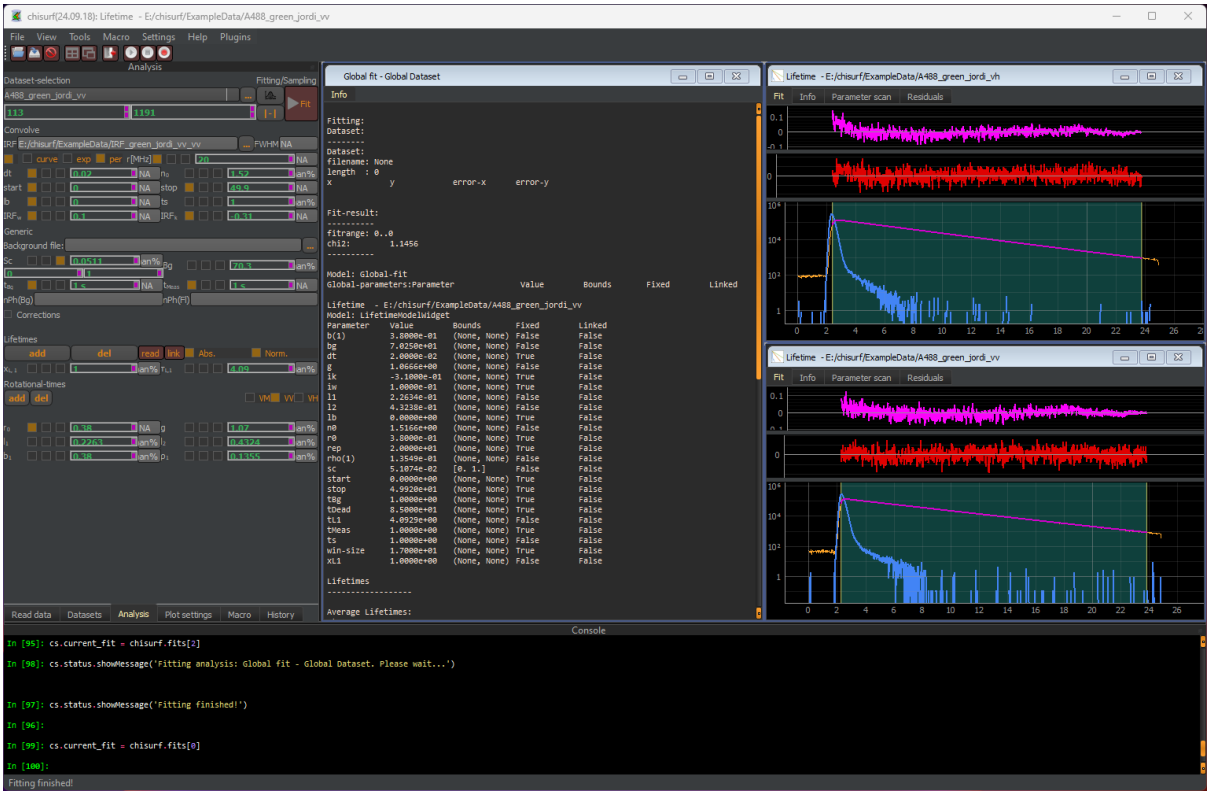
In this step, still the g-factor, l1 and l2 are fixed to the initially estimated values. Next, also perform a quick fit on the VH dataset. Important note: The fit will not be good! This only serves to initialize the parameter.

Step 2: Joint analysis to determine g-factor, l_s and l_p

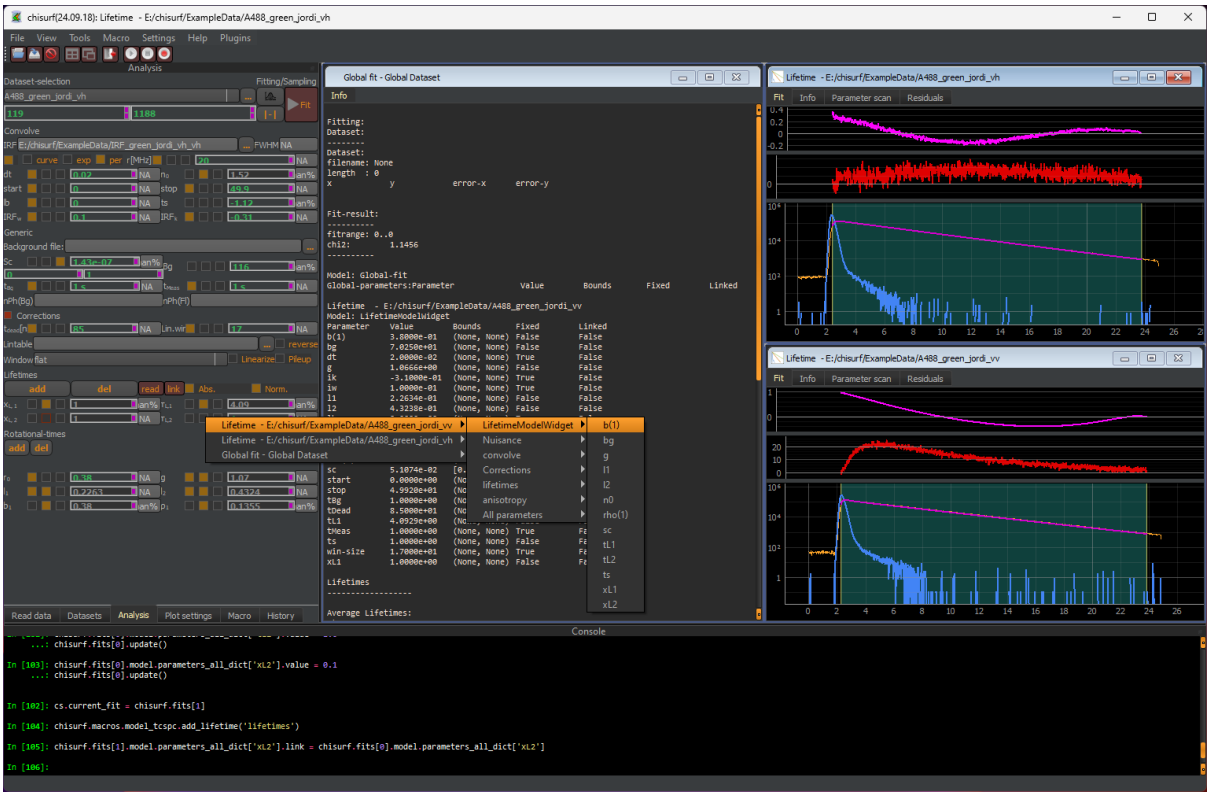
In the next step, the g-factor, l1 and l2 are no longer kept fixed to the initially estimated values but will also be fit and their respective check boxes are unticked.



Next, switch to the “Global fit” and press the fit button for the joint fit.



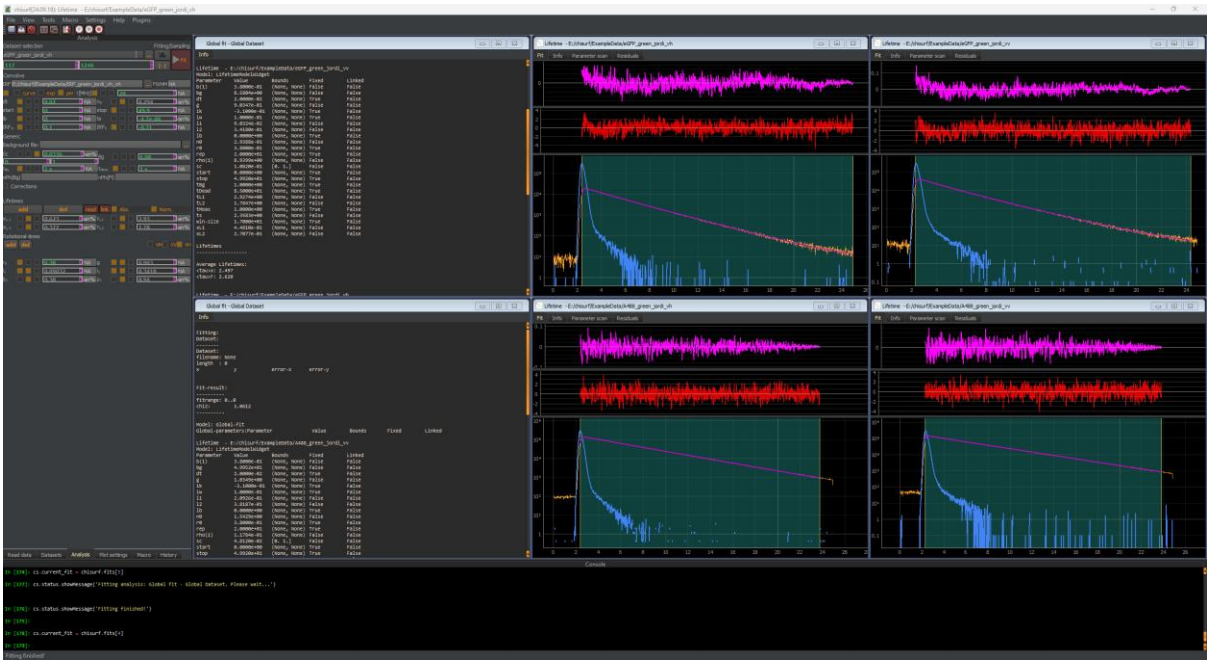
The fit is not yet good. It seems there is a component missing the in beginning of the decay. A second lifetime can be added and linked between VV and VH.



Once the modifications have been done to the VV and VH datasets, switch back to the “Global fit” and press the update button before performing a global fit again. Now, the fit looks good with flat residuals and a decaying autocorrelation function.

Loading and setting up the eGFP data set

Repeat all steps from above with the eGFP data set. Once completed, six windows should be present in ChiSurf.

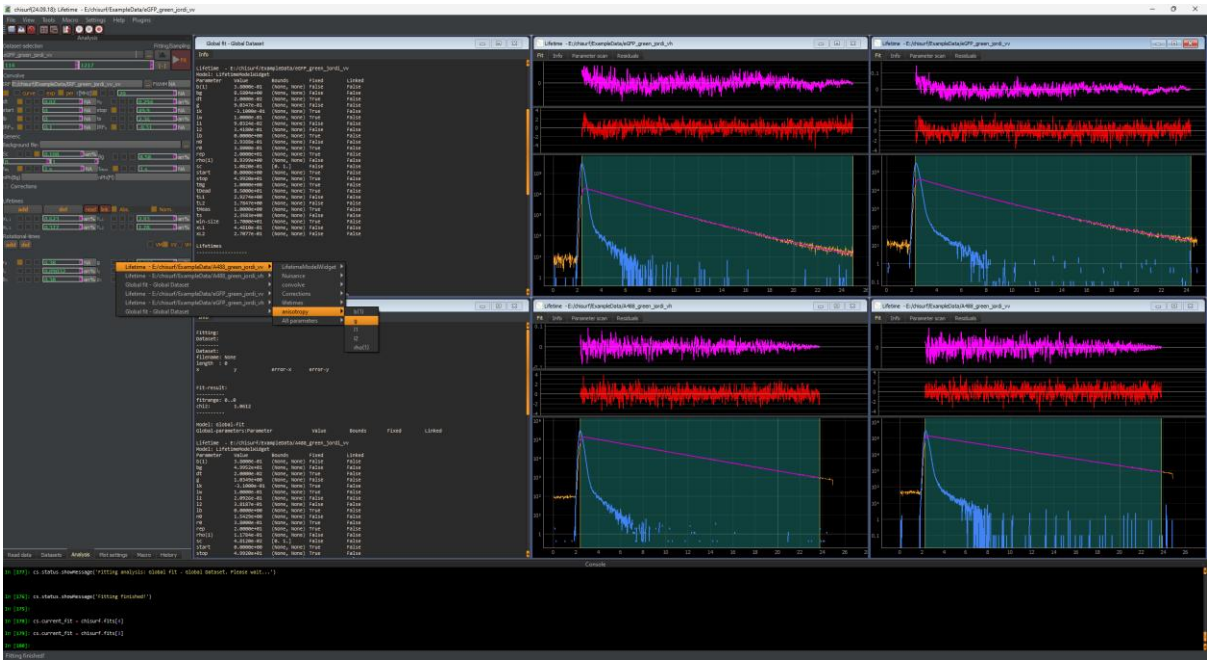


Note that as initial parameter for eGFP two lifetimes of 1.3 and 2.6 ns (~ 30:70) should be given, and a single rotational correlation time of 15 ns.

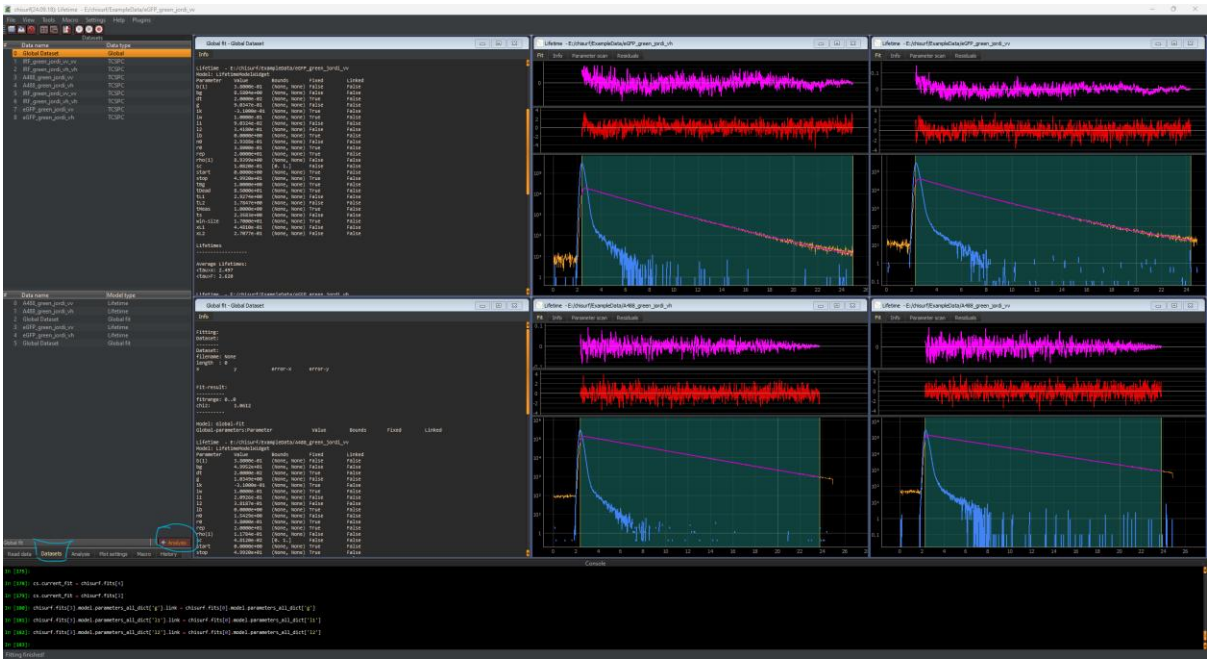
Setting up the joint global fit of A488 & eGFP

In the next step, a joint global fit of all four data sets (A488 VV, A488 VH, eGFP VV and eGFP VH) will be performed to best estimate the g-factor and the polarization correction factors.

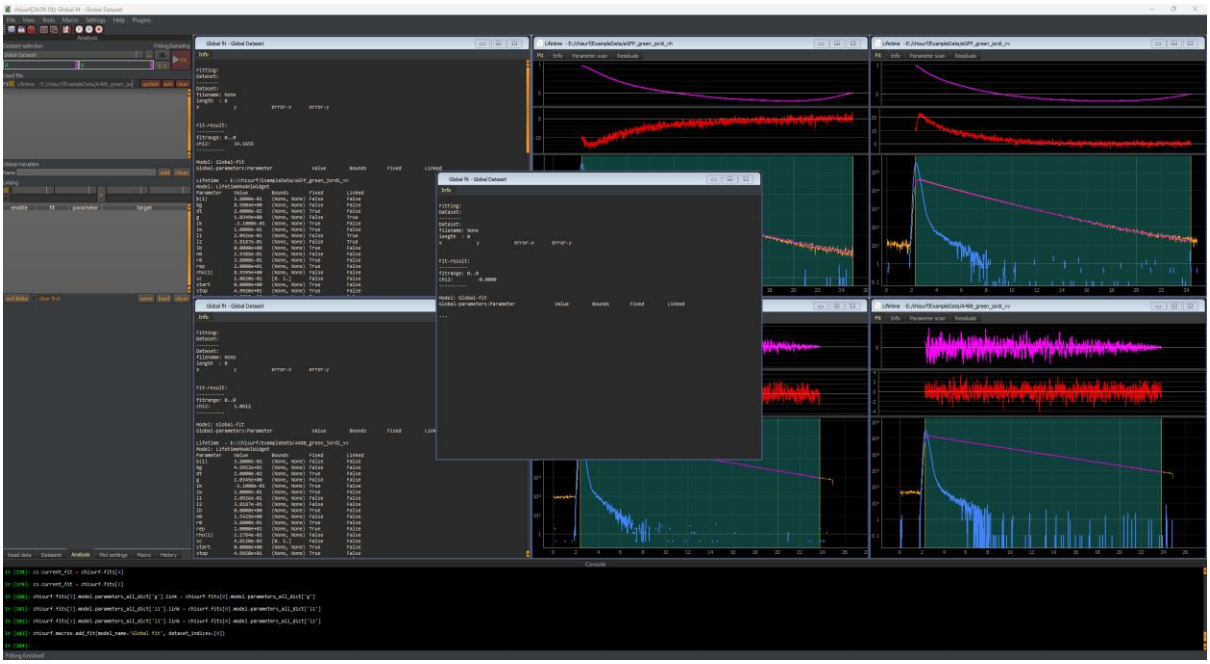
Firstly, the *g-factor*, *l1* and *l2* from eGFP VV will be linked to the respective parameter from A488 VV. Note these parameters in A488 VH and eGFP VH are already linked to the respective VV dataset, A488 VV or eGFP VV.



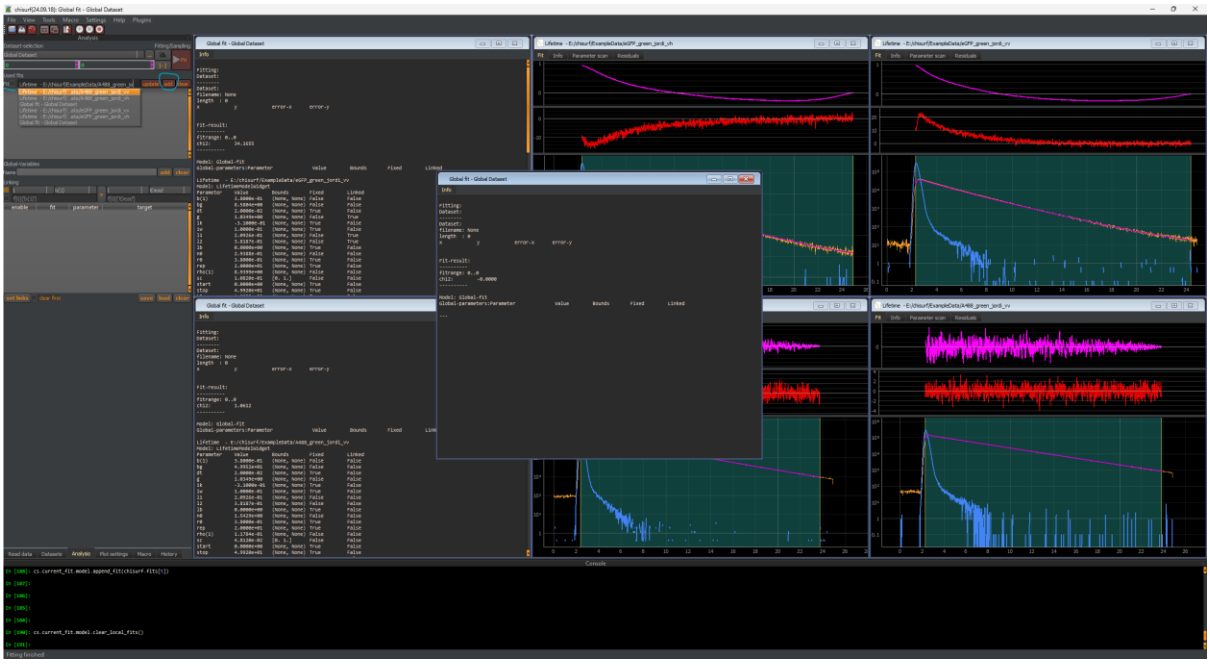
Next, a new global fit is added. Switch to the “Datasets” tab, select “Global Dataset” and press the “+ Analysis” button.



A new window will open.



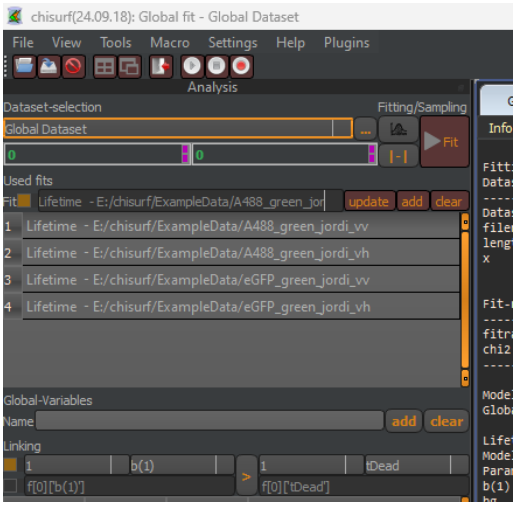
Switch back to the analysis tap and add the four relevant datasets for the joint fit to the global analysis by using the drop-down menu and the “add” button in the top left.



All four datasets need to be added:

- A488 VV
- A488 VH
- eGFP VV
- eGFP VH

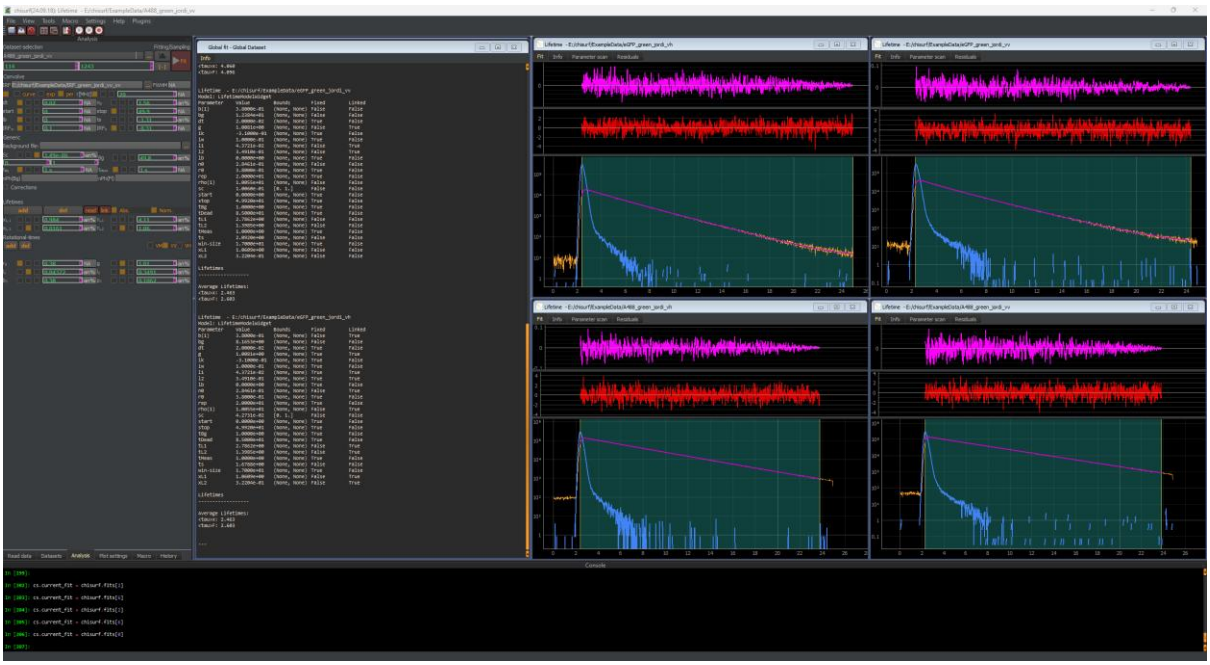
Do not add the previous global fits!



Finally, press the fit button and obtain the estimates for the g-factor and the polarization correction factor based on both data sets. Here, the following values are obtained:

- g-factor = 1.01
- l1 = 0.0437
- l2 = 0.349

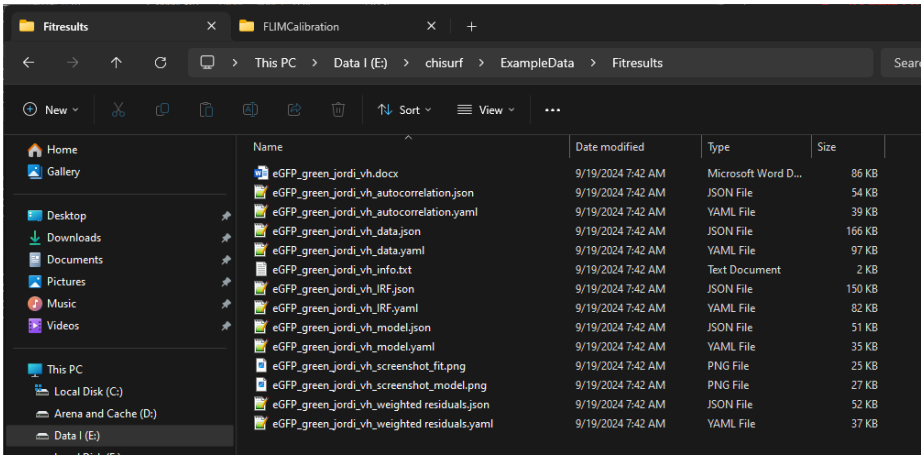
@Elizabeth: We have observed that the depolarization in one polarization direction is much larger in our used 40X/NA1.2 Zeiss water objective. This might be different for other objectives.



Saving & exporting the results

To save the results, either click on the respective fit window and press “Ctrl+S” or select from the main toobar at the top “File” -> “Save Fit results” -> “Current Fit”.

For each fit in total 14 files are saved:



- The word-file summarizes the results in a table.
- The *json* and *yaml* files are required to open the fit results again in ChiSurf.
- The wo png files are screen shots of the fit window and analysis window.
- The Info.txt summarizes the fit results table-like in a txt file.

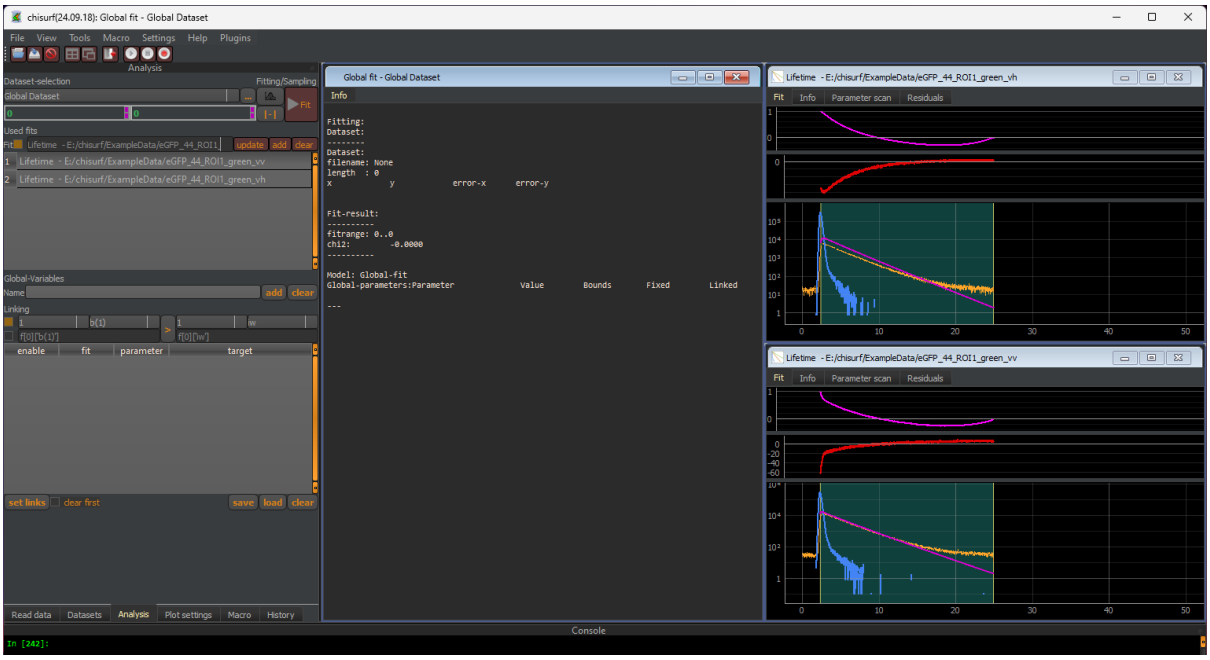
Re-using the parameter to analyse the actual samples

The obtained parameter can now be used to analyse the actual cell samples. Here, the example data stems from HEK293T cell transfected with cytosolic eGFP.

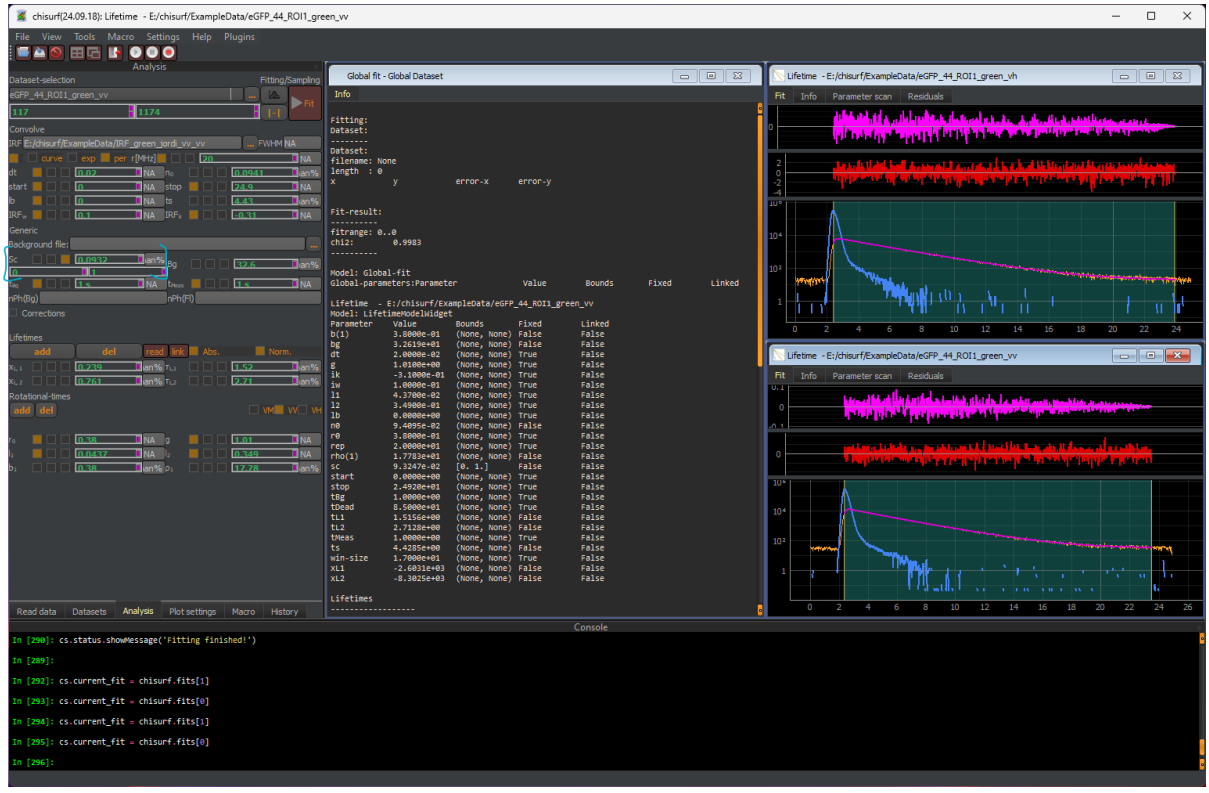
Firstly, verify that the reading parameter are still set correctly. Next, open the anisotropy wizard and load the cell data.

In the correction factor window, enter the obtained values and press save. In the fluorescence lifetimes and rotational correlation times window, the following components are given:

- Fluorescence Lifetime: 2.6 ns / 1.6 ns (0.7, 0.3)
- Rotational correlation times: 10 ns (0.38)



If the estimated parameter given in the fit setup are sufficiently close, one can directly proceed to the global fit window and press “Fit”.



Attention! Verify that no negative scatter fraction is present. Short-lived scatter might often e mixed up with fast rotational components.

Here we obtain an average eGFP lifetime of 2.44 ns and a rotational component of 17.8 ns.