PAM Documentation

Release 1.2

Anders Barth, Waldemar Schrimpf, Jelle Hendrix, Don C. Lamb

Jul 26, 2023

Contents

PAM is short for *PIE Analysis with MATLAB* - it is a GUI-based software package for the analysis of fluorescence experiments and supports a large number of analysis methods. While originally designed for *pulsed-interleaved excitation* (PIE, [Müller2005]) experiments, it now has a wide support for the most commonly used fluorescence experiments.

Currently implemented:

- Fluorescence correlation spectroscopy and its derivatives
  - FCS/FCCS/FLCS/fFCS
- Single-molecule FRET spectroscopy (burst analysis), including:
  - Photon distribution analysis for two- and three-color FRET experiments
  - species-selective filtered fluorescence correlation spectroscopy
- Image correlation spectroscopy, including:
  - ICS/RICS/TICS/STICS
  - ratser lifetime image correlation spectroscopy (RLICS)
  - support for arbitrary region image correlation spectroscopy (ARICS)
- Fluorescence lifetime imaging (PhasorFLIM)
- TCSPC and time-resolved anisotropy analysis

CHAPTER 1

Getting Started

Contents

In this part, we try to give a brief step-by-step walkthrough for the installation and usage of the main functions of *PAM* and its sub-modules. In most cases, the walkthrough will be based on the example data provided at https://gitlab.com/PAM-PIE/PAM-sampledata. More advanced methods and settings are explained in other parts of the manual.

1.1 Installing PAM

The program can be downloaded from the homepage of Prof. Don C. Lamb - http://www.cup.uni-muenchen.de/pc/lamb/ - or through *GitLab* - https://gitlab.com/PAM-PIE.

This *GitLab* group contains four projects:

- **PAM:** The open-source version of PAM used with MATLAB.
- **PAMcompiled:** A compiled stand-alone version that does not require MATLAB.
- **PAM-sampledata:** Some example data the user can use to try out the PAM functionalities.
- **PAMdocs:** Project that contains files for the developers to update the PAM documentation and manual.

1.1.1 Installing the stand-alone version of PAM

1. Verify the correct MATLAB Runtime is installed (currently v9.1/ Matlab 2016b).

   > You can download the MATLAB Runtime from the MathWorks Web site by navigating to
   >
   > http://www.mathworks.com/products/compiler/mcr/index.html
   >
   > For more information about the MATLAB Runtime and the MATLAB Runtime installer, see Package and Distribute in the MATLAB Compiler documentation in the MathWorks Documentation Center.

2. Download the compiled version of PAM for your Operating system (MacOS or WIN) from

   > https://gitlab.com/PAM-PIE/PAMcompiled

3. Unpack the files.
4. Run the *PAM.exe* (Windows) or *run_PAM.command* (MacOS) to start the program. For MacOS, consult the *readme.txt* and *how_to_create_shortcut.txt* for additional information.

1.1.2 Installing and updating the open-source version of PAM

The open source version of PAM requires a valid licence for MATLAB (2014b or newer). Certain features further need access to tool boxes (curve fitting, image processing, optimization, statistics and machine learning, parallel computing) to work.

You can obtain and update **PAM** either through direct download from *Gitlab*, using the command line through *Git*, or by using the MATLAB *Git* integration.

Downloading from the repository

1. Download the open source version of PAM from https://gitlab.com/PAM-PIE/PAM
2. Unpack the files.
3. Start Matlab and navigate to the PAM folder.
4. Type *PAM* into the Matlab command line to start the program.
5. To update, simply download the newest version and overwrite your files.

Using Git

Installing Git

Updating of **PAM** requires the installation of *Git*.

- **MacOS:**

  MacOS has *Git* pre-installed since version 10.9. Try to run *git* from the terminal. If the command fails, you can download *Git* from https://git-scm.com.

- **Windows:**

  Download and install *Git* from https://git-for-windows.github.io.

Downloading and updating using Git

- **To clone the repository, type *git clone https://gitlab.com/PAM-PIE/PAM.git PAM* to create a local copy of the repository in t**

  - By default, you are checkout on the branch *master*, containing the stable version of **PAM**.
  - Should you need the newest updates, switch to the *develop* branch by typing *git checkout develop*.
  - To switch back, type *git checkout master*.

- To update, simply type *git pull* to obtain the newest changes. Note that this will only update the branch that you are currently on.

Via MATLAB Git Integration

To use all features of the MATLAB *Git* integration, it is recommended to install *Git* as described above.

1. Create a folder for PAM.
2. Start Matlab and navigate to the PAM folder.

3. Right click the ‘Current Folder’ panel in Matlab and select ‘Source Control’ and ‘Manage Files…’.
4. Set the ‘Source control integration’ to ‘Git’ and enter for the ‘Repository path’ https://gitlab.com/PAM-PIE/PAM.git.
5. Click ‘Retrieve’ to download the files automatically.
6. To update the PAM Git repository to the latest version, simply type *!git pull* into the MATLAB command line.

To update, you can alternatively use the MATLAB *Git* integration, however the *!git pull* method is easier and quicker.

1. Right click the ‘Current Folder’ panel in Matlab and select ‘Source Control’ and ‘Fetch’.
2. Right click the ‘Current Folder’ panel in Matlab and select ‘Source Control’ and ‘Manage Branches’.
3. **Select the current branch form the remote repository, called ‘origin’.** If you work on branch *master*, select *origin/master* (or *refs/remotes/origin/master*).
4. Click ‘Merge’ to merge the changes from the remote into you local repository.

1.1.3 Modification and development of PAM

Users are encouraged to modify the code of PAM for their individual needs and help with the development of new and the improvement of existing functionalities.

To report bugs or suggest improvements and bugfixes, please use the Issues function of GitLab. For this, a free GitLab account is required.

To contribute to PAM, please consult the contribution guide.

1.2 Where to find the sample data

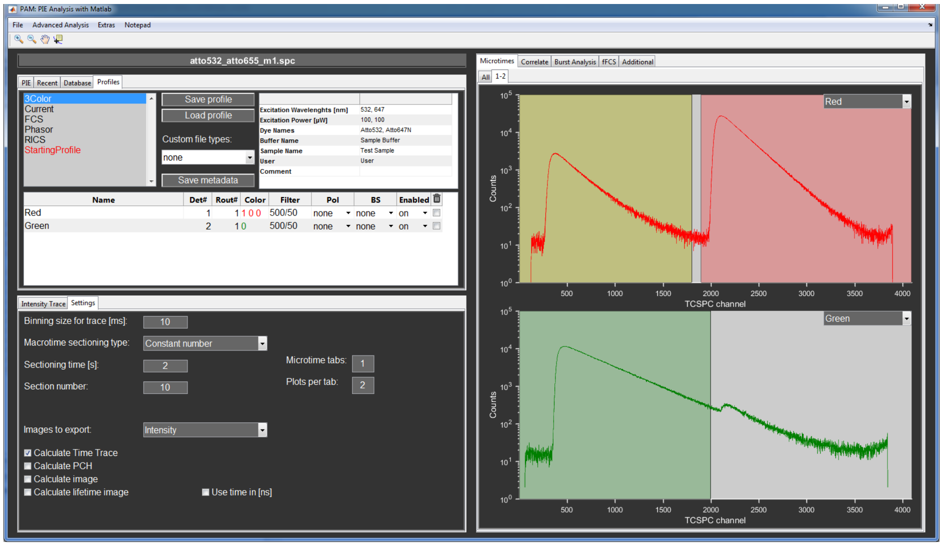
The sample data is provided at https://gitlab.com/PAM-PIE/PAM-sampledata, including profiles for the different setups. Copy the profile files (*.mat*) into the */profiles* folder of your PAM installation and restart PAM to make them available.

1.3 Setting up PAM

PAM is designed as a program running in the MATLAB environment. The used functionalities require a licensed MATLAB version 2014b or later and several toolboxes (optimization, statistics and imaging). A compiled standalone version that does not require a MATLAB installation will be available, but with limited functionalities especially for data export and custom plotting, since no access to the raw data is available via global variables. To install **PAM**, simply download the .zip file and extract it. Open MATLAB and navigate to the **PAM** folder. It is advised not to add the **PAM** folder to the Matlab path, as this might create confusion later on when using concurrent **PAM** versions. Here, type *PAM* into the command window to start **PAM**, or type the name of a sub-programs to start the particular window directly. Alternatively, *right-click* the respective *.m* file and select *Run*.

1.3.1 Setting up a Profile

All settings in PAM are defined in a user profile that can be selected in the *Profiles list* in the top-left corner of the *Profiles* tab in **PAM**. A minimal starting profile is automatically created, if none exists. How to create a new profile is described for the example of a two-color PIE microscope with individual detectors and B&H SPC150 TCSPC cards for green and red. The sample data is found in the */SetUp* folder.


PAM: PIE Analysis with Matlab

File Advanced Analysis Extras Notepad

atto532_atto655_m1.spc

PIE | Recent | Database | Profiles

3Color
Current
FCS
Phasor
RICS
StartingProfile

Save profile

Load profile

Custom file types: none

Save metadata

Excitation Wavelenghts [nm]	532, 647
Excitation Power [µW]	100, 100
Dye Names	Atto532, Atto647N
Buffer Name	Sample Buffer
Sample Name	Test Sample
User	User
Comment	

Name	Det#	Rout#	Color	Filter	Pol	BS	Enabled	
Red	1	1	1 0 0	500/50	none	none	on	□
Green	2	1	0	500/50	none	none	on	□

Intensity Trace | Settings

Binning size for trace [ms]: 10

Macrotime sectioning type: Constant number

Sectioning time [s]: 2

Section number: 10

Microtime tabs: 1

Plots per tab: 2

Images to export: Intensity

☑ Calculate Time Trace
□ Calculate PCH
□ Calculate image
□ Calculate lifetime image
□ Use time in [ns]

Fig. 1: Overview of the Profile and Settings configured for the example explained below.

Right click the *Profiles list* and select *Add new profile* to create a new profile.

A popup will open asking for a name for the profile. Click *Ok* once you typed in the name.

Click on the new profile in the *Profile list* and press *enter* or *right click* and select *Select profile*.

The new profile now only has one detector channel. In our case, we need a second channel for the red detector.

Alternative 1: *Right click* on the *Detector list* and select *Add new microtime channel*. We now have to set the *Det#* value of the second channel to 2, corresponding to our second TCSPC card.

Alternative 2: **PAM** has an automatic channel detection feature. When data is loaded that is not associated with an existing channel, a new channel will be created automatically. To turn this feature off/on, *right click* the *Detector list* and select *Auto-detect used detectors and routing*

Load the data set *atto532_atto655_m1.spc* from the */SetUp* folder using the *Multi-card B&H SPC files recorded with B&H-Software* routine.

You can change the color and name associated with the channels as you like (e.g. Green and Red).

To display the microtime histogram of both channels, we adjust the *Microtime tabs* and *Plots per tab* properties in the *Settings* tab. In this case, we set the *Plots per tab* to two. Now, the histograms are shown in the *Microtimes* tab.

Now, we need to define PIE channels. Most calculations and functions of **PAM** are based on these PIE channels.

Initially, a single PIE channel associated with the first detector channel is created. In our example, we need to create two more. For this, *right click* the *PIE channel list* and select *PIE channel* and *Add new PIE channel*.

For the two-color PIE setup, we need one PIE channel for the green detection after green excitation (**GG**), one for red detection after red excitation (**RR**) and a third for red detection after green excitation (**GR**). The last combination possible PIE channel of green detection after red excitation (**RG**) does not contain useful signal and can be neglected.

Set the associated detector channel of the **GR** and **RR** channels to the red detector and **GG** to the green.

The microtime ranges associated with each laser depend heavily on each individual setup. In this case we assume that the green laser pulse arrives in the first half of the microtime range and the red in the second half. The total range is 4096 TCSPC channels.

Set the range for the **GG** and **GR** channels to 1-2048 and the **RR** channel to 2049-4096.

You can also change the color associated with each channel by *right clicking* the *PIE channel list* and selecting *PIE channel -> Change channel color*, or via the *c* key.

The different PIE channels should now also be indicated in the microtime histograms as colored backgrounds. Make sure that they overlap with the corresponding decays.

1.4 FCS and FCCS

1.4.1 Performing FCS calculations

To perform FCS and FCCS calculations, first load your data. You can also load multiple files at once. The data will then be concatenated and treated as a single, longer measurement.

In this case, we will show how to perform correlations based on the *atto532_atto655* B&H *.spc* example data. For this, select the *FRET-FCS* profile. Here, we have a total of 4 detectors, for green parallel, green perpendicular, red parallel and red perpendicular.

First load the data. For this, you need to select the *_m1.spc* format (B&H multi-card format).

To select on which particular PIE channels you want to perform the auto- and cross-correlation, use the checkbox table in the *Correlate* tab. The diagonal elements represent the auto-correlations. The others correspond to the cross-correlation of the row and the column PIE channels.

For this example, we will calculate the auto-correlation for the green and the red channels plus the cross-correlation between them.

For the auto-correlations we will correlate **GG1** with **GG2** and **RR1** with **RR2**. By calculating the correlation from signal of two different detectors and TCSPC cards, we can remove correlation artifacts from detector afterpulsing and TCSPC dead time. This is especially noticeable at very short delay times (~100ns). Since the dyes are freely rotating, any polarization effects can be neglected in this case.

For the cross-correlation, we will use the combined channels to improve the photon statistics. For green, this is the **GG1+GG2** PIE channel. For red, we will use the **RR1+RR2** channel. We can also do a cross-correlation between the green channel and the full red channel (**GR1+GR2+RR1+RR2**). This represents the case of not using PIE.

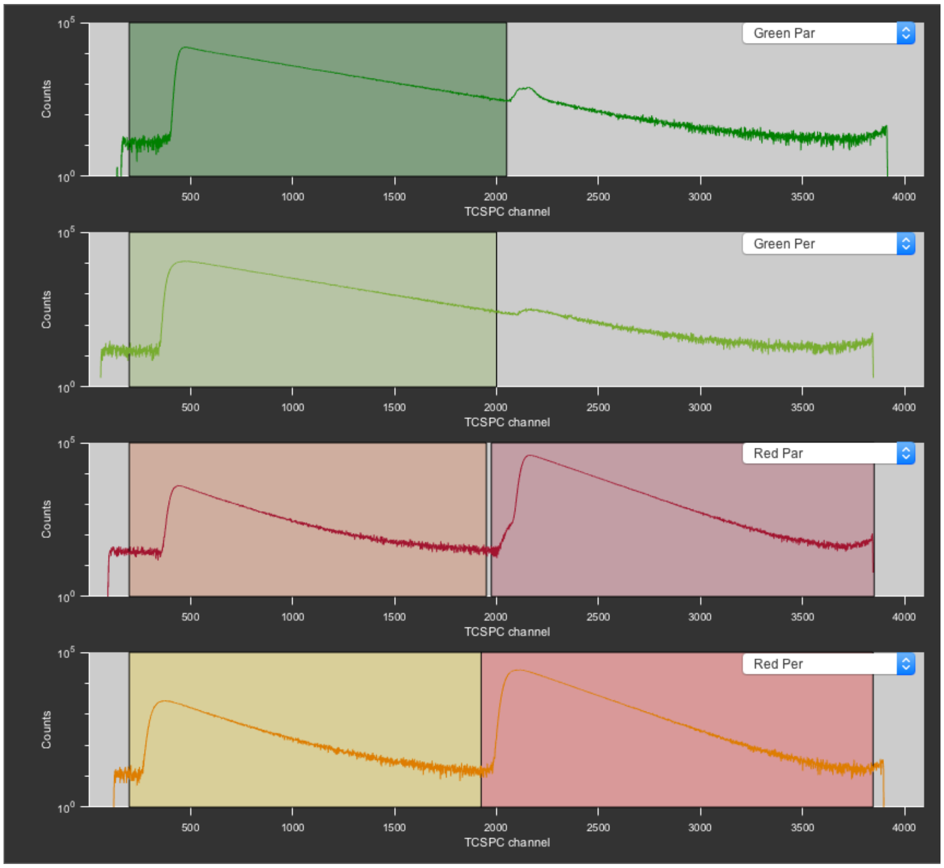


Fig. 2: The PIE channels defined for the FCS measurement. Top to bottom: GG1, GG2, GR1 and RR1, GR2 and RR2.

Next, select the type of correlation (for standard FCS use *Point*) and the file type. The *.mcor* format is a **PAM** specific type based on the standard Matlab data format (*.mat*) and can be easily loaded into Matlab (for plotting etc.). The *.cor* format is a simple text file that can be used to look at the data outside of Matlab or **PAM**. Both can be analyzed with the **FCSFit** sub-module.

FCS calculations are done in individual segments, indicated by the gray segments in the *intensity trace* plot. You can set either the total number of segments or their length with the *Section time* or the *Section number* properties in the *Settings* tab.

You can remove individual segments from the calculations by simply *left clicking* on them. This is useful to remove small artifacts, like aggregates, air-bubbles or dust, from the correlation.

To start the correlation, simply press the *Correlate* button in the *Correlate* tab. The files are saved automatically in the folder of the raw data with the same filename and an added suffix of 'channelname1_x_channelname2'.

Once the calculation is finished, the correlation curves will be shown in the bottom right plot. Each selected PIE channel combination is displayed in an individual plot. The individual curves represent the selected segments. They can be discarded individually with a *left click* on the curves. If you choose to unselect segments at this point, *right click* the plot and select *Save selected correlations* to update the file to you new selection.

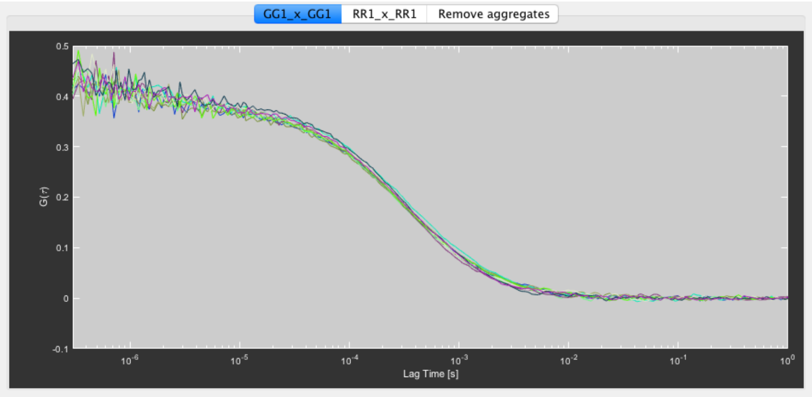


Fig. 3: The preview plot of the individual segments after the correlation has been performed.

Multiple files can be correlated separately by clicking the *Multiple* button in the *Correlate* tab. A file selection dialog will open and the user can select multiple file to be correlated. Each selected file is loaded and correlated separately one after another. The same settings (PIE channels, selected segments, etc.) are used for all of them.

Another way to automatically perform multiple correlations is done using the database functions.

1.4.2 Using FCSFit

FCSFit is the sub-module of **PAM** for analyzing FCS and FCCS data. It can be opened via the *Advanced Analysis* menu in the menu bar of *PAM* or from the command window by typing in *FCSFit*.

Here we will show its functions on the case of the data used for the FCS part (*atto532_atto655*).

Once you opened the window, first load the fit model you want to use. Select *File* and *Load Fit Function*. The last fit model used during the previous session will be preloaded. For this example, load the *FCS_TauD.txt* model. This is a simple diffusion through a 3D gaussian focus model with an added triplet dynamics term.

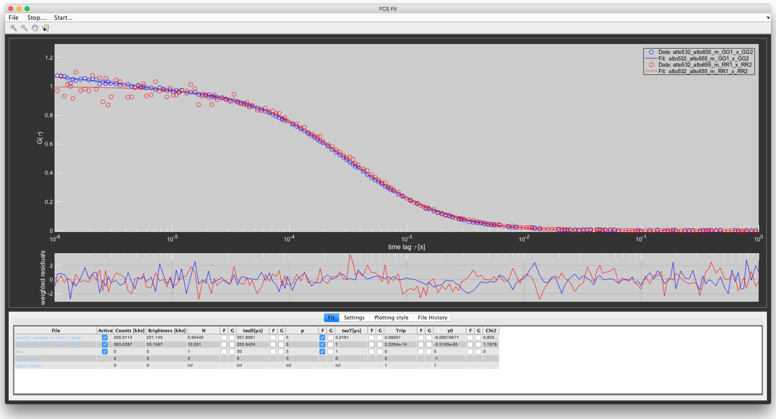
Next, load the correlation files. *File -> Add Files* will add the new files to previously loaded once, while *Load New Files* will clear the previous entries.

The fit parameters can be adjusted in the *Fit parameters* table in the *Fit* tab at the bottom. The *F* anf *G* checkboxes can be used to fix or globally link individual parameters, respectively.

The settings for the fitting procedure can be adjusted in the *Settings* tab.

Select *Start…Fit* in the menu bar to begin the fit.

Once the fit is finished, the results will be displayed in the *Fit parameters* table.

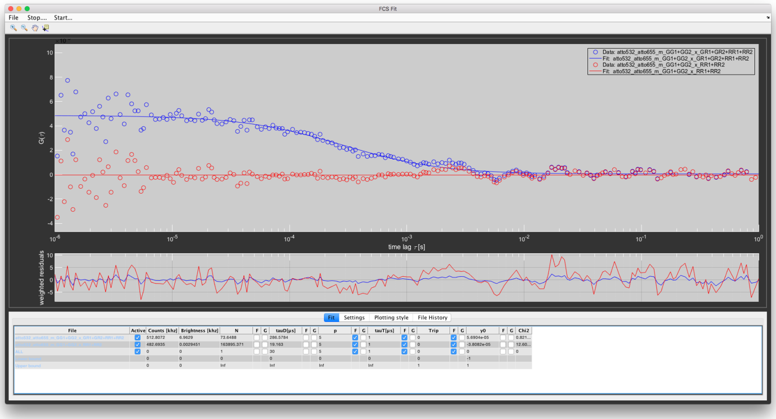


File	Active	Counts [khz]	Brightness [khz]	N	F	G	tauD[µs]	F	G	p	F	G	tauT[µs]	F	G	Trip	F	G	y0	F	G	Chi2
atto532_atto655_m_GG1_x_GG2	☑	200.0114	221.149	0.90442	□	□	307.8061	□	□	5	☑	□	3.0191	□	□	0.08597	□	□	-0.00019671	□	□	0.803...
atto532_atto655_m_RR1_x_RR2	☑	283.2287	23.1567	12.231	□	□	325.9424	□	□	5	☑	□	1	□	□	2.2204e-14	□	□	-2.5193e-05	□	□	1.1378
ALL	☑	0	0	1	□	□	30	□	□	5	☑	□	1	□	□	0	□	□	0	□	□	0
Lower bound		0	0	0			0			0			0			0			-1			
Upper bound		0	0	Inf			Inf			Inf			Inf			1			1			

Fig. 4: Fits to the autocorrelation functions GG1xGG2 and RR1xRR2.

The fit results can be copied to the clipboard with the *Copy Results to Clipboard* options accessed by *right-clicking* the *Fit parameters* table.

The FCS curves and fits can also be exported to a new Matlab figure with a *right click* on the plot and selecting *Export to figure*. The export settings can also found in the *Settings* tab.


FCS Fit
File
Stop....
Start...
Data: atto532_atto655_m_GG1+GG2_x_GR1+GR2+RR1+RR2
Fit: atto532_atto655_m_GG1+GG2_x_GR1+GR2+RR1+RR2
Data: atto532_atto655_m_GG1+GG2_x_RR1+RR2
Fit: atto532_atto655_m_GG1+GG2_x_RR1+RR2
G(τ)
10
8
6
4
2
0
-2
-4
10^{-6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}
10^{-1}
10^{0}
time lag τ[s]
weighted residuals
10
5
0
-5
Fit
Settings
Plotting style
File History

File	Active	Counts [khz]	Brightness [khz]	N	F	G	tauD[µs]	F	G	p	F	G	tauT[µs]	F	G	Trip	F	G	y0	F	G	Chi2
atto532_atto655_m_GG1+GG2_x_GR1+GR2+RR1+RR2	☑	512.8072	6.9629	73.6488	□	□	286.5784	□	□	5	☑	□	1	☑	□	0	☑	□	5.6904e-05	□	□	0.821...
atto532_atto655_m_GG1+GG2_x_RR1+RR2	☑	482.6935	0.0029451	163895.371	□	□	19.163	□	□	5	☑	□	1	☑	□	0	☑	□	-3.8082e-05	□	□	12.60...
ALL	☑	0	0	1	□	□	30	□	□	5	☑	□	1	☑	□	0	☑	□	0	□	□	0
Lower bound		0	0	0			0			0			0			0			-1			
Upper bound		0	0	Inf			Inf			Inf			Inf			1			1			

Fig. 5: Fits to the cross-correlation functions (GG1+GG2)x(RR1+RR2) - i.e. using PIE, and (GG1+GG2)x(GR1+GR2+RR1+RR2) - i.e. not using PIE. When PIE is not used, the spectral cross-talk of the green dye leads to residual cross-correlation amplitude.

1.5 Single-molecule FRET

For this tutorial, use the *FRET-FCS-profile* and load the sample data *DNA_1h* using the option *Multi-card B&H SPC files recorded with B&H-Software*. The data folder additionally contains measurements of the buffer and a pure water sample to measure the instrument response function (IRF) for lifetime fitting. The profile already contains the data from these measurements (background photon counts, IRF pattern), so we do not need to load these data files here. See the section in the PAM manual for a description of how to define these calibrations.

1.5.1 Extracting bursts for single-molecule FRET

The loaded data was recorded using PIE and polarized detection, resulting in for detection channels and a total of 6 *PIE channels* (**GG1**, **GG2**, **GR1**, **GR2**, **RR1**, **RR2**), which are already set up in the profile.

Switch to the *Burst Analysis* module in **PAM**. Select *Sliding Time Window* as **Smoothing Method** and *APBS 2C-MFD* as the **Burst Search Method**. This will perform an *all-photon burst search* (*APBS*), pooling all photons of the individual detection channels into one, using the *sliding time window* burst search. First, we need to assign *PIE channels* to the *FRET channels* used in the burst analysis. *FRET channels* are denoted by **DD** for the donor signal after donor excitation, **DA** for the FRET signal after donor excitation, and **AA** for the acceptor signal after direct excitation by the acceptor laser, split into parallel and perpendicular polarization. Assign the PIE channel **GG1** to **DD**-parallel, **GG2** to **DD**-perpendicular, **GR1** to **DA**-parallel, **GR2** to **DA**-perpendicular, **RR1** to **AA**-parallel and **RR2** to **AA**-perpendicular. Now, set the burst search parameters as 500 μ s for the time window, 100 photons minimum per burst and a threshold of 5 photons per time window (i.e. 10 kHz count rate threshold).

Click the *Preview* button to see a preview of the burst search. Use the arrow to navigate through the preview plot.

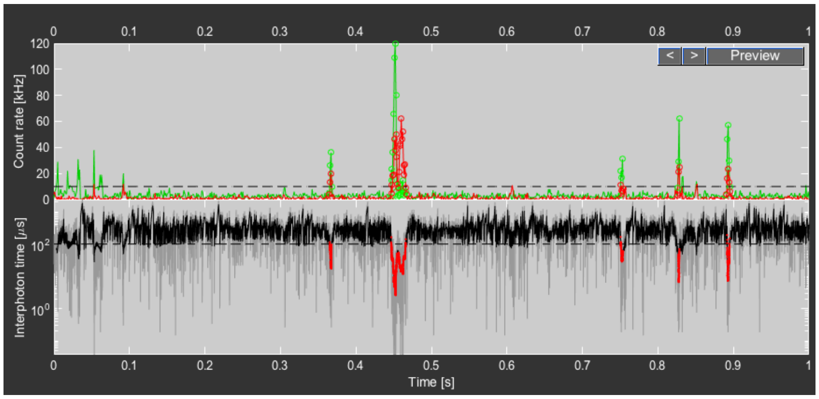
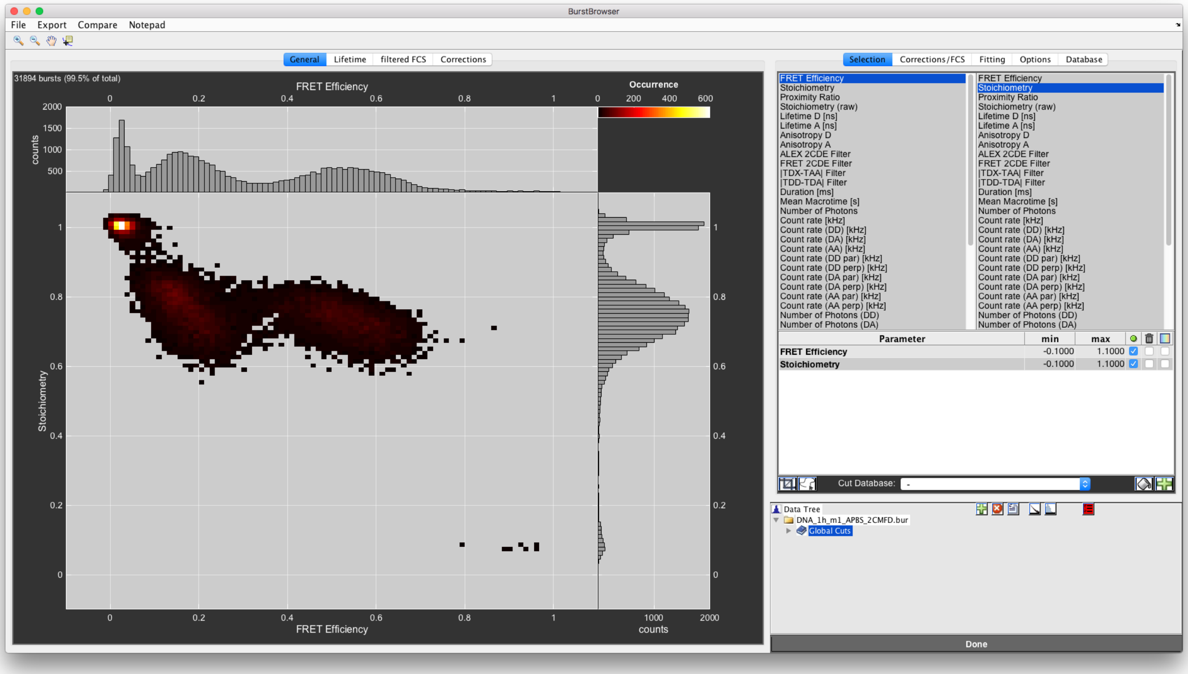


Fig. 6: Preview of the burst search.

Press the *Do Burst Search* button to start the burst search. This will perform the burst search and store the result in a *.bur* file using the filename of the raw data in a sub-folder with the name of the raw data. After the burst search has been performed, the *Do Burst Search* button will be green. For a description of how to determine the *burstwise fluorescence lifetime* and the *two channel kernel density estimation filter* (*2CDE*), see the respective section in the *PAM manual*.

1.5.2 Using Burst Browser

Open **BurstBrowser** using the menu *Advanced Analysis -> BurstBrowser* or by typing *BurstBrowser* in the command-line. Click *File -> Load New Burst Data* and select the file *DNA_1h_m1_APBS_2CMFD.bur* in the *DNA_1h_m1* subfolder.



31894 bursts (99.5% of total)

Parameter	min	max	●	🗑	▨
FRET Efficiency	-0.1000	1.1000	☑	□	□
Stoichiometry	-0.1000	1.1000	☑	□	□

Cut Database: -

Fig. 7: The unprocessed burst analysis result shows two FRET populations.

You can change the plotted parameters using the lists in the top right. The left list selects the parameter for the x-axis, the right parameter for the y-axis. *Right-click* any parameter to add it to the *Cut Table*, where you can specify lower and upper bounds to constrain the data selection.

In this case, we want to exclude the donor-only and acceptor-only populations. To do so, set restrict the range for the *Stoichiometry* in the *Cut Table* to the interval [0.3,0.9]. This will remove contributions of donor- and acceptor-only molecules to the FRET efficiency histogram completely.

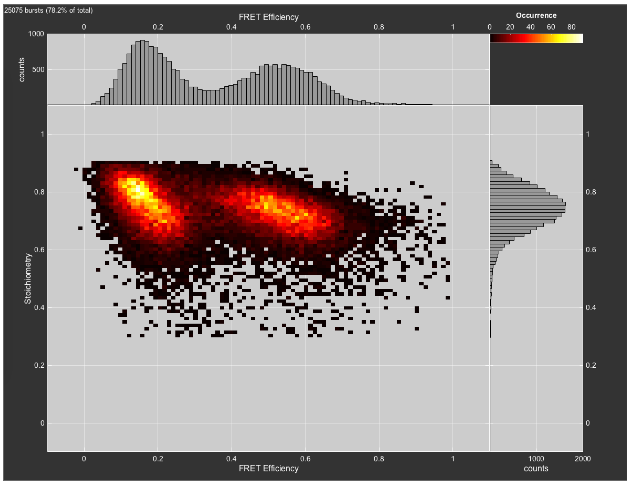


Fig. 8: Filtered burst data after removal of donor- and acceptor-only molecules through the stoichiometry parameter.

For a more detailed description of the functionalities of **BurstBrowser** see the *manual*.

1.6 Image Analysis

The key steps of using the image analysis functions of **PAM** will be explained with the example of the *RICS_EGPF.ppf* file. For this, select the *RICS* profile first.

1.6.1 Exporting image data from PAM

There are two ways to export the image data of the individual PIE channels from **PAM**.

The first one is to load the data into **PAM**.

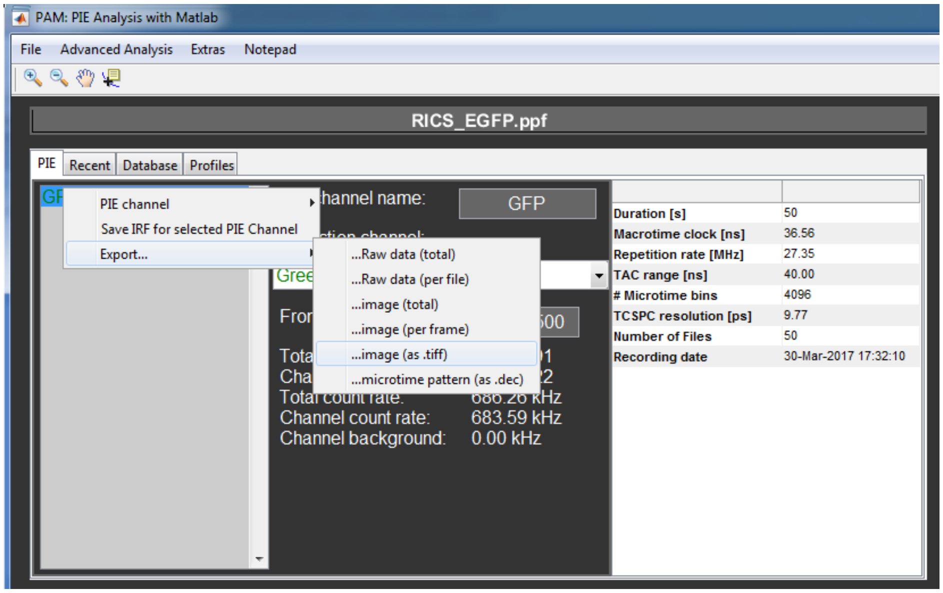
Next, select the PIE channels in the *PIE channel* list for which you want to export the data.

Right click the list and select *Export..* and *..image (as .tiff)*.

Select the folder in which you want the new image files to be saved.

For each selected PIE channel a separate file will be created with the raw data file name and the PIE channel name as a suffix.

The second way of exporting TIFFs from **PAM** uses the database functions and is explained *here*.



Channel name: GFP

Total count rate: 686.26 kHz
Channel count rate: 683.59 kHz
Channel background: 0.00 kHz

Duration [s]	50
Macrotime clock [ns]	36.56
Repetition rate [MHz]	27.35
TAC range [ns]	40.00
# Microtime bins	4096
TCSPC resolution [ps]	9.77
Number of Files	50
Recording date	30-Mar-2017 17:32:10

Fig. 9: Exporting images as TIFF stacks.

1.6.2 Using MIA

MIA is the image analysis module of **PAM**. It can be started from **PAM** via *Advanced Analysis* or from the command line by typing in *Mia*.

Loading data into MIA

MIA can use two types of data.

The first uses the image data currently loaded in **PAM**.

To load this data, use the *Load…* and *… data from PAM* functions in **MIA**. The PIE channels used for this are defined in the *Channels* tab with the *PIE Channels to load* options.

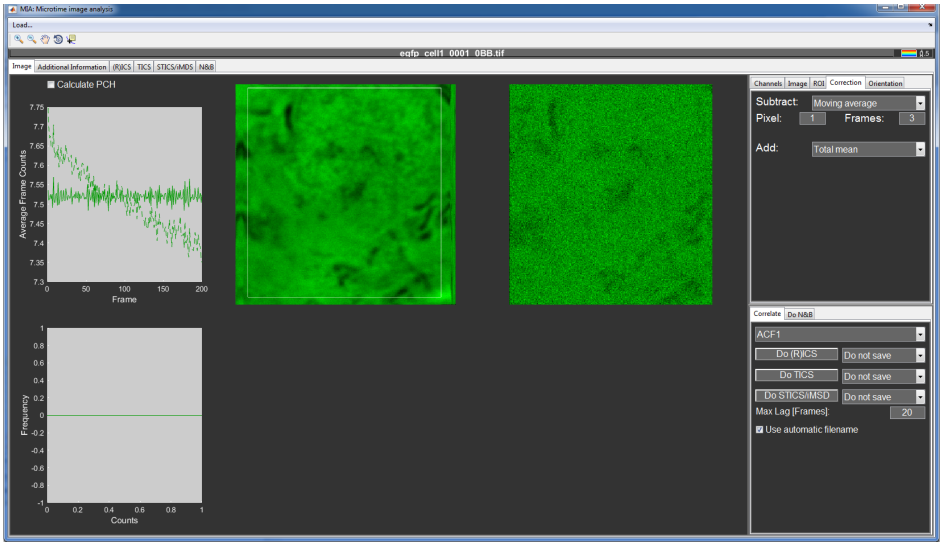
The more common type of data is based of TIFFs. Generally, **MIA** can use any type of data that can be converted to TIFFs, either with **PAM** (explained above) or other programs.

To load new TIFFs, use the *Load…* and *single color TIFFs* functions in **MIA**.

Next, select your desired TIFF files for the first channel and click *Open*. When multiple files are selected, they will be simply concatenated.

A new file selection will open up, asking you to select the files for the second channel. Click *Cancel* if you only want to analyze single color/channel data.

The data will them be loaded and displayed.



□ Calculate PCH

Subtract: Moving average
Pixel: 1 Frames: 3

Add: Total mean

Do (R)ICS: Do not save
Do TICS: Do not save
Do STICS/iMSD: Do not save
Max Lag [Frames]: 20
☑ Use automatic filename

Fig. 10: Single color data loaded into MIA.

Performing image correlations in MIA

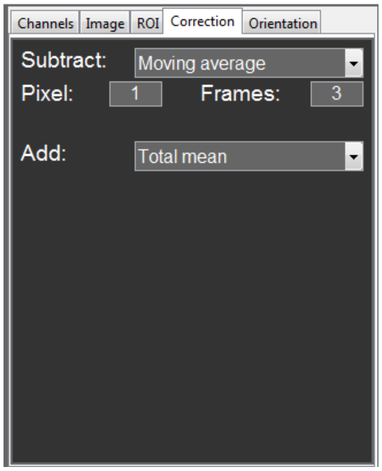
To do any type of analysis in **MIA**, fist load the data.

Select a subregion of the loaded file in the left image plots of **MIA**. This can be done with the mouse in the left images or with the edit boxes in the *ROI* tab.

An advanced region of interest selection can be employed by selecting the *Arbitrary region ICS* settings in the *ROI* tab.

The images can then be further adjusted in the *Correction* tab. This can be useful, e.g. to remove immobile structures for RICS analysis or bleaching effects for TICS.

For the *RICS_EGFP* data, select a moving average of 1 Pixel and 3 Frames to be subtracted and add the total mean to keep the mean intensity constant.



Subtract: Moving average

Pixel: 1 Frames: 3

Add: Total mean

Fig. 11: Standard correction settings for RICS analysis.

This removes the correlation from immobile structures in the cell and also corrects for the slight bleaching.

Now that the images are corrected, you can calculate the image correlation by pressing the *Do (R)ICS* button.

To save the data, choose a file format before performing the correlation. *.miacor* is a format based on the Matlab data format and can easily be loaded into the MATLAB workspace at a later point in time. You can also save the data as TIFFs to look at them with other image analysis programs. Both file formats can be loaded into **MIAFit**.

With the standard settings, the program creates a *MIA* folder in the folder of the raw data and saved the correlation files automatically. For the filename, the program used the original name with a suffix for the correlation channel (ACF1, ACF2 or CCF). Additionally, a info file is generated that contains the key settings used.

You can activate a manual file naming by unselecting the *Use automatic filename* checkbox.

You can have a quick look at the correlations in the *(R)ICS* tab, but the main way of analysis is performed with the sub-module **MIAFit**.

1.6.3 Using MIAFit

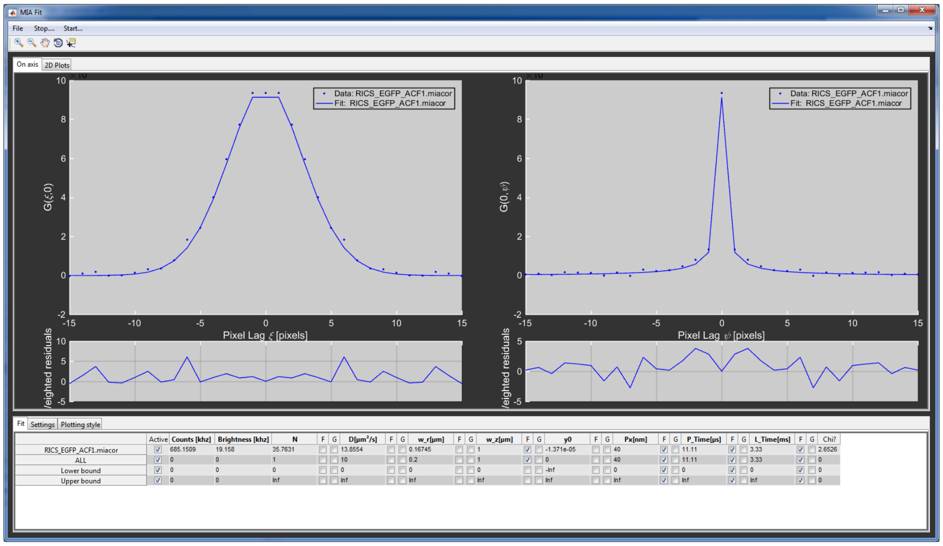
MIAFit can be started from the *Advanced Analysis* menu in **PAM** or by typing **MIAFit** into the matlab command line.

MIAFit works basically the same as *Using FCSFit*.

The difference is in the way the 2D data is displayed.

You can switch between on-axis view for all files (*On-Axis* tab) or a full 2D correlation display for a single file (*2D Plots*);

For the *RICS_EGFP* file, load the *RICS_Simple3D* model. Set the pixel size (*Px*) to 40 nm, the pixel time (P_time) to 11.11 μs and the line time (L_time) to 3.33 ms and fix all these values. It is generally useful to fix the axial focus size (*w_z*), because this parameter is often hard to determine with RICS. Set it to 1 μm.



	Active	Counts [khz]	Brightness [khz]	N	F	G	D[µm²/s]	F	G	w_r[µm]	F	G	w_z[µm]	F	G	y0	F	G	Px[nm]	F	G	P_Time[µs]	F	G	L_Time[ms]	F	G	Chi?
RICS_EGFP_ACF1.miacor	☑	685.1509	19.158	35.7631	☐	☐	13.8554	☐	☐	0.16745	☐	☐	1	☑	☐	-1.371e-05	☐	☐	40	☑	☐	11.11	☑	☐	3.33	☑	☐	2.6526
ALL	☑	0	0	1	☐	☐	10	☐	☐	0.2	☐	☐	1	☑	☐	0	☐	☐	40	☑	☐	11.11	☑	☐	3.33	☑	☐	0
Lower bound	☑	0	0	0	☐	☐	0	☐	☐	0	☐	☐	0	☐	☐	-Inf	☐	☐	0	☑	☐	0	☑	☐	0	☑	☐	0
Upper bound	☑	0	0	Inf	☐	☐	Inf	☐	☐	Inf	☐	☐	Inf	☐	☐	Inf	☐	☐	Inf	☑	☐	Inf	☑	☐	Inf	☑	☐	0

Fig. 12: MIAFit RICS analysis window.

1.7 Phasor approach to FLIM

1.7.1 Calculating lifetime phasor data

How to calculate phasor data from you raw data file is explained for the example of the sample phasor data provided with the download. The data is based on the *PAM* simulation file format.

First, switch to the *Phasor* profile. This profile is configured for a single detector channel and a single PIE channel.

Load a reference file (*Phasor_Reference_4ns.sim*). Remember that you have to select the *PAM simulation file (.sim)* format in the file selection menu.

Go to the *Phasor Referencing* tab in the *Additional* tab.

Since this example only has one channel, the correct channel is already selected. In other cases you might have to first select the correct detector channel from the dropdown menu. Phasor referencing works directly on the detector channels and is independent of the PIE channels

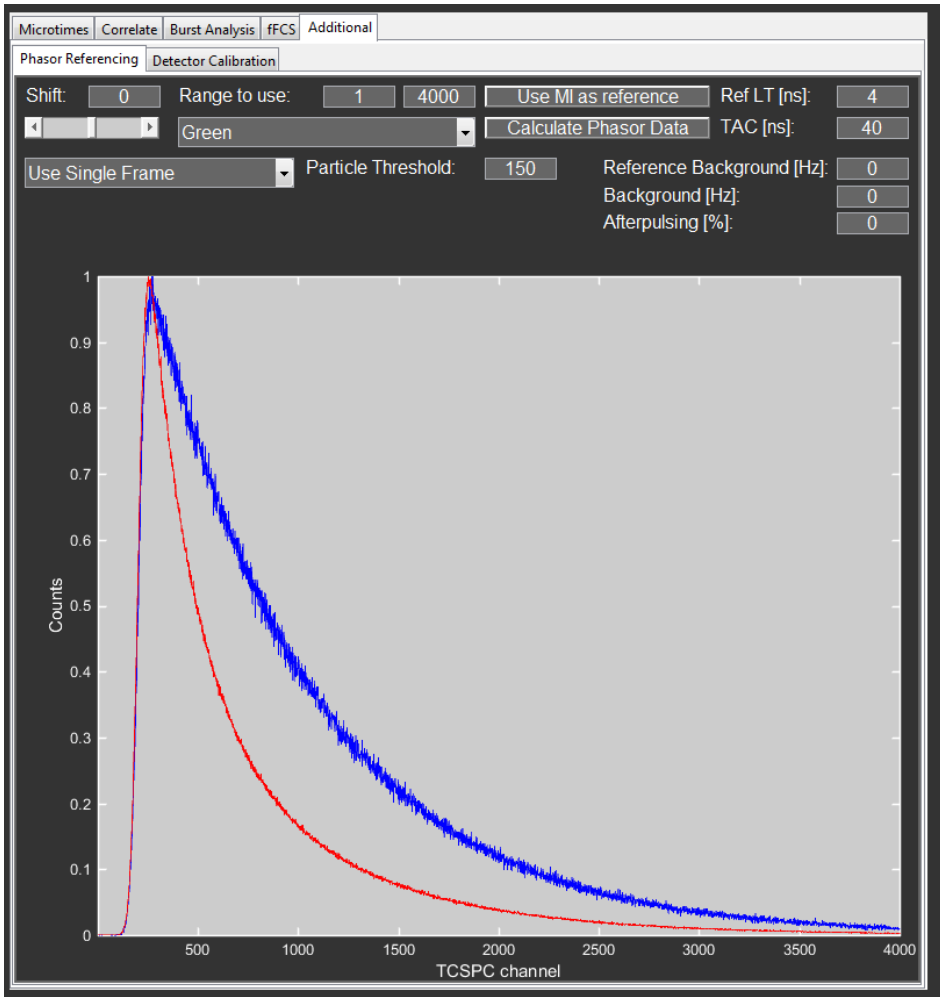
Define the loaded file as the reference by clicking the *Use MI as reference* button. Now, this microtime histogram will be used as the reference file until it is replaced, even across sessions.

Next, load the files you want to use (*Phasor_Sample.sim*) for the phasor calculations.

Use the *Range to use* fields to adjust the microtime range used for the calculation to your data. For example, if you recorded two color PIE data and want to analyze the red lifetime, remove the green cross-talk or FRET part from the range. The program uses the full microtime range, but sets all values outside of the defined range to zero. In this case you can use the full range of 1-8192.

Before performing the calculation, set the *Ref LT* field to the lifetime of your reference (here 4ns) and the *TAC* field to the duration of the full microtime range (here 40ns).

To calculate the phasor and save the data, click the *Calculate Phasor Data* button. Select a path and filename and click *Save* to save the data.



Shift: 0

Range to use: 1 4000

Green

Use Single Frame

Particle Threshold: 150

Ref LT [ns]: 4

TAC [ns]: 40

Reference Background [Hz]: 0

Background [Hz]: 0

Afterpulsing [%]: 0

Fig. 13: Phasor Referencing setup for the provided sample data.

1.7.2 Phasor Analysis

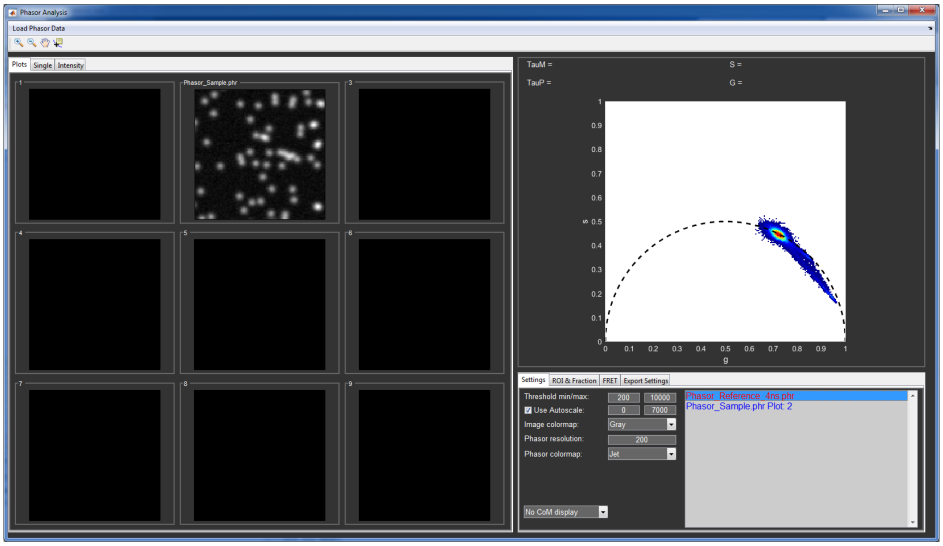
The main features of the phasor analysis sub-module are explained for the example of the sample phasor data provided with the download.

Open the **Phasor** analysis window from the *Advanced Analysis* menu in **PAM** or by typing in *Phasor* into the MATLAB command line.

Use *Load Phasor Data* to load files you created with **PAM** or the one provided with the download (*Phasor_Reference_4ns.phr* and *Phasor_Sample.phr*).

The intensity images are now shown to the nine image plots at the left hand side of the window. The files are also added to the *Phasor file list* in the bottom right and the phasor histogram is plotted in the top right.

You can (de)activate individual files by selecting them in the *Phasor file list* and using with the *arrow* keys or remove them with the *del* key. The + key will set the file to inactive, meaning that it will be used for calculating the phasor histogram, but not shown in the image plots.



TauM =
S =
TauP =
G =

Threshold min/max: 200 10000
☑ Use Autoscale: 0 7000
Image colormap: Gray
Phasor resolution: 200
Phasor colormap: Jet

Phasor_Reference_4ns.phr
Phasor_Sample.phr Plot: 2

Fig. 14: Phasor analysis window of the provided sample data. The first file in the list is shown in red, meaning that it is in the inactive state.

The *threshold* values remove pixels with intensity below or above the set values from the phasor histogram. You can change them to see how it affects the histogram.

The other parameters in the *Settings* tab mainly affect how the data is displayed.

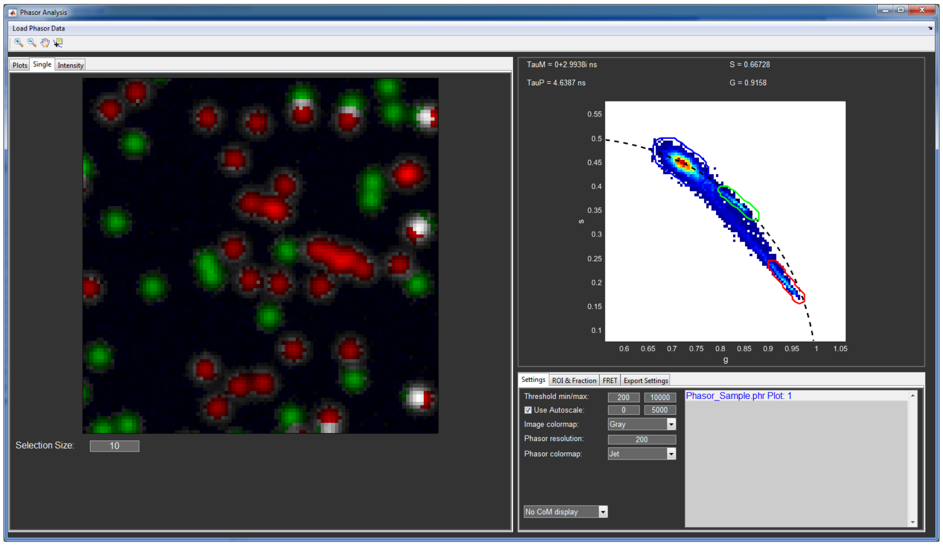
In order to indicate the phasor position in the images, there are two principle ways:

Method 1: Region of interest.

Press and hold the number keys *1-6* and click into the phasor histogram to select a region of interest. The *left mouse button* creates a rectangular and the *right mouse button* a ellipsoid ROI.

The pixels inside this ROI will now be assigned the color associated with the corresponding ROI

You can change the width of the ellipsoid and the color in the *ROI&Fraction* tab.



TauM = 0+2.9938i ns
S = 0.66728
TauP = 4.6387 ns
G = 0.9158

Selection Size: 10

Threshold min/max: 200 10000
☑ Use Autoscale: 0 5000
Image colormap: Gray
Phasor resolution: 200
Phasor colormap: Jet

Phasor_Sample.phr Plot: 1

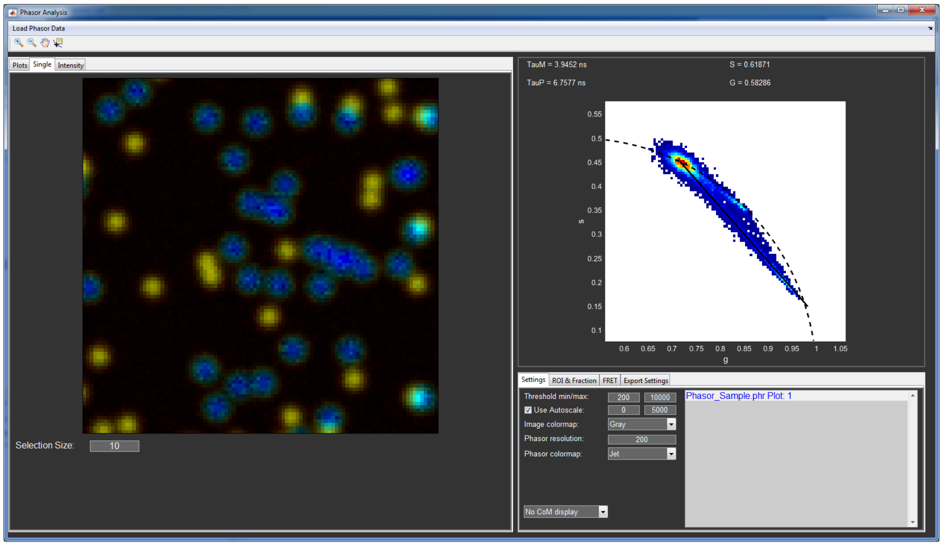
Fig. 15: ROI selection for *Phasor* analysis

Method 2: Position indication.

Press and hold the number keys *0* and click into the phasor histogram to place a line into the phasor histogram.

The pixels will be assigned a color corresponding to the closes point on this line.

You can change the maximal distance from the line and the color-code use in the *ROI&Fraction* tab.



TauM = 3.9452 ns

TauP = 6.7577 ns

S = 0.61871

G = 0.58286

Selection Size: 10

Threshold min/max: 200 10000

☑ Use Autoscale: 0 5000

Image colormap: Gray

Phasor resolution: 200

Phasor colormap: Jet

No CoM display

Phasor_Sample.phr Plot: 1

Fig. 16: Line selection for *Phasor* analysis

Use the *mouse wheel* to zoom into the phasor histogram and the *left mouse button* to move the view.

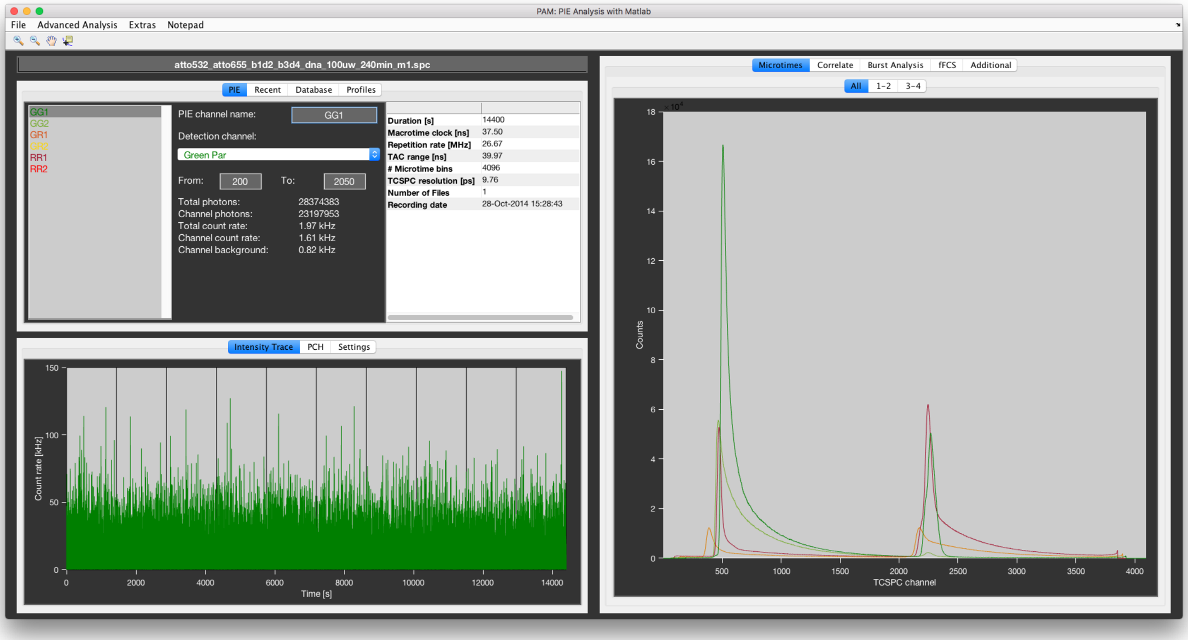
Right click the images to export them.

Double click the phasor histogram to export it to a new figure. The setting for this can be adjusted in the *Export Settings* tab.

CHAPTER 2

PAM - PIE Analysis with MATLAB

Contents



atto532_atto655_b1d2_b3d4_dna_100uw_240min_m1.spc

PIE channel name: GG1

Detection channel: Green Par

From: 200 To: 2050

Total photons: 28374383
Channel photons: 23197953
Total count rate: 1.97 kHz
Channel count rate: 1.61 kHz
Channel background: 0.82 kHz

Duration [s]	14400
Macrotime clock [ns]	37.50
Repetition rate [MHz]	26.67
TAC range [ns]	39.97
# Microtime bins	4096
TCSPC resolution [ps]	9.76
Number of Files	1
Recording date	28-Oct-2014 15:28:43

Fig. 1: PAM main gui.

2.1 Basic functionalities

2.1.1 Loading data

Load data by choosing *File -> Load TCSPC data* in the menu bar. Currently, **PAM** supports most TCSPC data file formats. Becker&Hickl hardware is supported for cards of type 130, 140, 150, 630 and 830 (*.spc* files). PicoQuant hardware is supported for the TimeHarp200, TimeHarp260(N/P), PicoHarp and HydraHarp (*.ptu*, *.ht3*, *.t3r* files). For PicoQuant only the *T3* data format is supported because of the data handling in PAM, which is based on separated macro- and microtime information. (The *T2* format records photon arrival time with maximum TCSPC resolution in one array, which can be reconstructed from the separate macro- and microtime photon streams in software. We don't recommend to use this file format, since information is lost compared to the *T3* format.)

While **PAM** is primarily designed for TCSPC data, many of its features (e.g. FCS or image fluctuation analysis) also work with certain non-TCSPC counting or pre-binned data. The way to deal with this type of data is described in more detail in *Data structure* and *Custom file formats*.

Standard formats

The following read-in options are always available:

Single-card B&H SPC files recorded with B&H-Software (*.spc*): This is the general read-in routine for Becker&Hickl (B&H) *.spc* files. Choose this if you want to load data from a single SPC card.

Multi-card B&H SPC files recorded with B&H-Software (*._m1.spc*): Use this read-in routine if you are recording data on multiple SPC cards simultaneously. When using multiple SPC cards with the B&H software, the data will be saved in individual files for each card, numbered by the extension *_m1.spc*, *_m2.spc* and so forth. Using this read-in routine, select the first *.spc* file, and all associated files with same name but different number will be loaded.

PicoQuant universal file format (*.ptu*): PicoQuant's general time-tagged time-resolved file format, which can be recorded by different hardware. Information about the used hardware is encoded in the file and read out upon loading.

HydraHarp400 TTTR file (*.ht3*): File format specific to the PicoQuant HydraHarp recorded using the version 2 software.

TimeHarp200 TTTR file (*.t3r*): File format specific to the TimeHarp200 card from PicoQuant.

PhotonHDF5 file (*.h5*): Photon HDF5 file as described on http://photon-hdf5.github.io.

PAM simulation file (*.sim*): Simulated data from **PAM**'s simulation module.

PAM photon file (Created by PAM) (*.ppf*): Raw photon data exported from **PAM**.

Confocor3 raw files (*.raw*): Raw data files recorded with the Confocor 3 module for Zeiss microscopes.

Custom file formats

The high number of different commercial and custom microscope makes it impossible to accommodate all data and file formats that could be used to analyze data in **PAM**. An especially big problem is the vast number of different way to encode scanning markers in the data, even when one of the standard TCSPC formats is used. Therefore, we designated a spot in the filetype selection for a custom read-in routine. These routines are stored in the Custom_Read_ins folder in the functions folder and must be matlab function files (*.m*) of a certain base structure. One of these read-ins can be selected in the **profiles** tab in **PAM** and is stored in the *UserValues* information. This way each lab can create and adjust local read-ins.

The read-in function must have the following function name:

```
function [Suffix, Description] = MyFunction(FileName, Path, Type, PAM_handle)
```

The input parameters are:

FileName : A *1* x *n* cell array containing a string with the filename of each file to be loaded

Path : A string of the folder path of the files

Type : The type must always be 10 and is used to encode the custom read-in information in the FileInfo

PAM_handle : A structure containing all the handles of GUI elements of the **PAM** main figure. This input does not need to be used in the function, but can be used to query certain setting in **PAM**.

The output parameters are strings used for the *Filterspecs* and *DialogTitle* inputs of the matlab *uigetfile* function.

The function name should then be followed by a short section like this:

```
if nargin == 0

    Suffix ='*.mytype';

    Description ='My custom data format (*.mytype)';

    return;

end
```

This section has the purpose to filter the file selection by a certain file ending and to give a short description of the datatype. Outside of this section the output parameters are not used anymore.

The last requirement of the custom read-in function is to declare some variables global using:

```
global TcspcData FileInfo
```

The rest of the code does not have to follow a certain form and can contain other functions or references to non-matlab based functionalities. The only requirement is that the global variables **TcspcData** and **FileInfo** have a specific form (described in *Data structure*).

2.1.2 Data structure

Most data in **PAM** is stored in the memory as global variables to allow the user easy access to extract the data at any point of the analysis. These variables are:

TcspcData

This variable contains the basic photon information. It is a *structure* with two fields, *MT* and *MI*. Both fields are *n* x *m* cell arrays, with *n* representing the different Tcspc cards or modules used, while *m* represents the routing information. These cells are addressed by the *Detector channels* definition of a profile.

Both *MT* and *MI* contain entries in these cells for each detected photon.

MT: The macro-time or time since the start of the experiment in clock ticks, stored as a *n* x *1 double* array. In most TCSPC devices the ticks are based on a internal or external clock, of which the period is stored in *FileInfo.ClockPeriod*. In the case of binned raw data, the clock period should be set to the bin duration and a pseudo-photon entry should be created for each intensity count.

MI: The micro-time or time since last synchronization pulse, stored as a *n* x *1 uint16* array. If non-TCSPC data is used, these entries should all be set to 1.

FileInfo

This variable contains all the information about the measurement required to perform proper analysis in **PAM**. It is a *structure* with the following required field:

FileType : Name of the datatype, needed to determine the correct the read-in routine if the data is to be re-loaded.

Type : Number defining the read-in routine. For custom read-ins the number is always 10.

FileName : Cell containing all the filenames as strings.

Path : Folder path of the loaded files

NumberOfFiles: Represents the number of currently loaded files

MI_Bins : Number of bines used for the micro-time

ClockPeriod : Define the period of the macro-time ticks

SyncPeriod : Defines the period between two laser- or synchronization pulses. Usually identical to the ClockPeriod. Currently not used.

TACRange : Defines the full range of the micro-time. Not always equal to ClockPeriod.

Lines : Number of line of a frame.

Pixels : Number of pixels per line.

MeasurementTime : Total time of the measurement in seconds

ImageTimes : Starting time of each frame in seconds. Has *Frames* or *Frames+1* entries (see ImageStops)

LineTimes : Starting time of each Line in seconds. Is an array of *Frames* x *Lines* or *Frames* x *(Lines+1)* (see LineStops)

LastPhoton : Arrival time of each photon of a file in clock ticks

ScanFreq : Scanning frequency for line or circular scans.

These fields are always required for **PAM** to work properly. When data is loaded that does not contain some of the information (e.g. the imaging parameters), they should be initialized to default parameters in the read-in routine.

If both start and stop signals are used for image data, two additional fields are used:

ImageStops : End time of each frame in seconds. If ImageStops is defined, both ImageStops and ImageTimes have *Frames* entries, otherwise ImageTimes has *Frames+1* entries.

LineStops : End time of each line in seconds. If LineStops is defined, both LineStops and LineTimes have *Frames* x *Lines* entries, otherwise LineTimes has *Frames* x *(Lines+1)* entries.

Additional fields can be created to store information for the user, but they will generally not be used by the program.

PamMeta

This variable contains pre-processed data and other information needed for a fast update of the plots.

UserValues

This variable contains a variety of different settings or other parameters (filepaths, IRF or phasor reference data, etc.) that are not specific to the current file and are stored between sessions. These data are saved in the profile and reloaded upon restart of the program or a change of the used profile.

2.1.3 Setting up a profile

PAM uses profiles to allow the users to work efficiently with data from different setups and configurations. A profile saves the settings of a session on the hard disk, so that the main parameters do not need to be re-entered every time.

Profile management is done in the **Profiles** tab in the tip-left of the **PAM** window. To switch to an existing profile, select it in the **profiles list** and press *enter* or right click and select **Select profile**. To create a new profile, right click the **profiles list** and select **Duplicate selected profile** or **Add new profile**. This will prompt the user to enter a name for the new profile. All profiles are stored in the profiles sub-folder of **PAM** as individual *.mat* files. When **PAM** is started, it checks this folder for the profiles list. Thus, profiles can be easily copied to and shared between different computers.

Hint: It is recommended to create a profile for each setup configuration to speed-up analysis of regular experiments.

Note: Switching profiles usually requires the user to reload the data into **PAM**. It is generally recommended to restart **PAM** when switching profiles to make sure that all settings are updated.

The two most important settings of a profile are the definitions of the detector and the PIE channels.

Detector channels

Data loaded into **PAM** is divided into separate channels for the different detectors used. Each channels is associated with an index for the TCSPC card or module (**Det#** in the **detector list**) and an index for the routing bit (**Rout#** in the **detector list**) used. These values correspond to the indexes of the *TcspcData* variable, where the corresponding data are stored.

To create a new detector channel, right click the **detector list** and select **Add new microtime channel**. The Name, indices and some additional information can be easily changed in the list.

Note: Consistent with the standard MATLAB convention, indexing of the channels/modules/routing starts at 1 and not 0, as with many other programming languages.

Hint: Individual detector channels can be temporarily disabled with the **Enabled** property in the **detector list**. For some data types (e.g. the B&H *.spc* files) this can speed-up data loading and save memory, if only a subset of the detectors is needed.

The individual detector channels are plotted in the *Microtime plots* and are directly used in some applications (e.g. Phasor referencing).

Defining PIE channels

The majority of applications in **PAM** do not work directly with the detector channels but rather use PIE channels. A PIE channel contains only photons from a part of the microtime range, thus it is possible to separate the data by excitation (in the case of pulsed interleaved excitation), or use time-gating.

The PIE channels are displayed in the **PIE list** of the **PIE** tab in the top-left of **PAM**. Right clicking the list opens a menu that allows the user to adjust the PIE channels (e.g. creating new channels or deleting existing ones). PIE channels are defined by the detection channel and the microtime range (in units of the microtime bins). The microtime range can be set manually using the edit boxes **To** and **From** in units of TAC channels, or interactively by *right-clicking* the PIE channel and choosing *PIE channel -> Manually select microtime range*. This will turn the cursor into a crosshair. Select the *From* and *To* points by clicking twice into the respective plot in the *individual microtime plot* tab.

PIE channel notation

Although you can give PIE channels any name you want, we recommend the following notation. Since the idea of PIE channels is to separate photons by detection color and excitation source, this information should be encoded in the given PIE channel name. Choose a single letter to encode the color (e.g. *G* for green, *R* for red, *B* for blue, *Y* for yellow) and use the single letter code to denote excitation and detection color in this order (e.g *GR*, *BY*, *BR*).

For example, if excitation was performed using a 532 nm laser (green, *G*), and the photons are detected in the red spectral region (red, *R*), the PIE channel for red photons after green excitation should be given the name *GR*. If additionally polarized detection was used, append a number to encode this information (1 for parallel polarization, 2 for perpendicular polarization).

Note: A typical PIE-MFD setup with two color excitation and two color polarized detection will thus have the following PIE channels: *GG1*, *GG2*, *GR1*, *GR2*, *RR1* and *RR2*.

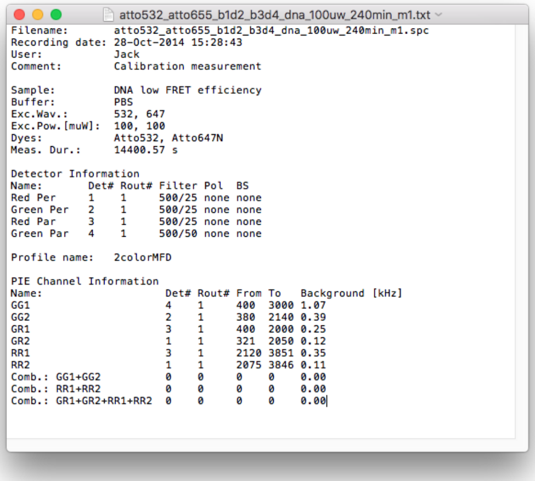
Combined PIE channels

In some cases it might be useful to pool data from multiple detectors (e.g. combine polarizations). Since each PIE channel is associated with only a single detector channel, this requires combined PIE channels. To create a combined PIE channel, select the individual ones, right click the *PIE channel list* and select *PIE channel -> Create combined channel*. The new channel will refer back to its parents, so that changes (e.g. of the microtime range) will be automatically updated.

Note: A combined channel uses the position of its parents in the list as reference. Deleting channels might thus disrupt the references. Therefore, combined channels should be re-defined whenever any changes were done to the PIE channel list.

Note: Most applications in **PAM** will treat combined channels just as any other channel. However, some functions (burst search, lifetime fitting and filtered FCS) require standard PIE channels to work.

Saving the meta data



```
Filename:       atto532_atto655_b1d2_b3d4_dna_100uw_240min_m1.spc
Recording date: 28-Oct-2014 15:28:43
User:           Jack
Comment:        Calibration measurement

Sample:         DNA low FRET efficiency
Buffer:         PBS
Exc.Wav.:       532, 647
Exc.Pow.[muW]:  100, 100
Dyes:           Atto532, Atto647N
Meas. Dur.:     14400.57 s

Detector Information
Name:       Det# Rout# Filter Pol  BS
Red Per     1    1     500/25 none none
Green Per   2    1     500/25 none none
Red Par     3    1     500/25 none none
Green Par   4    1     500/50 none none

Profile name:   2colorMFD

PIE Channel Information
Name:                   Det# Rout# From To   Background [kHz]
GG1                     4    1     400  3000 1.07
GG2                     2    1     380  2140 0.39
GR1                     3    1     400  2000 0.25
GR2                     1    1     321  2050 0.12
RR1                     3    1     2120 3851 0.35
RR2                     1    1     2075 3846 0.11
Comb.: GG1+GG2          0    0     0    0    0.00
Comb.: RR1+RR2          0    0     0    0    0.00
Comb.: GR1+GR2+RR1+RR2  0    0     0    0    0.00
```

Fig. 2: An example meta data file.

To save meta data associated with your measurement, enter the respective information into the table in the **Profiles** tab. Click **Save Metadata** to store the meta data in a *.txt.* file with the same name as the loaded file in the respective folder.

In addition to the user specified information from the meta data table, the recording date and measurement duration are stored together with the information from the detection channel table and the PIE channel definitions including the stored background count rates.

2.1.4 Data exploration

Several standard plots of the raw data are available to investigate the data set. *Microtime histograms* are plotted on the right side. On demand, a *intensity trace*, *photon counting histogram* or *image* can be plotted. Disable/enable these plots in the **Settings** tab using the respective checkboxes.

Hint: Disabling the calculation of *intensity trace*, *photon counting histogram* and/or *image* will significantly speed up the loading of data, especially for large data sets. Re-enabling the plots after the data has been loaded will still provide you with the plots on-demand.

Microtime plots

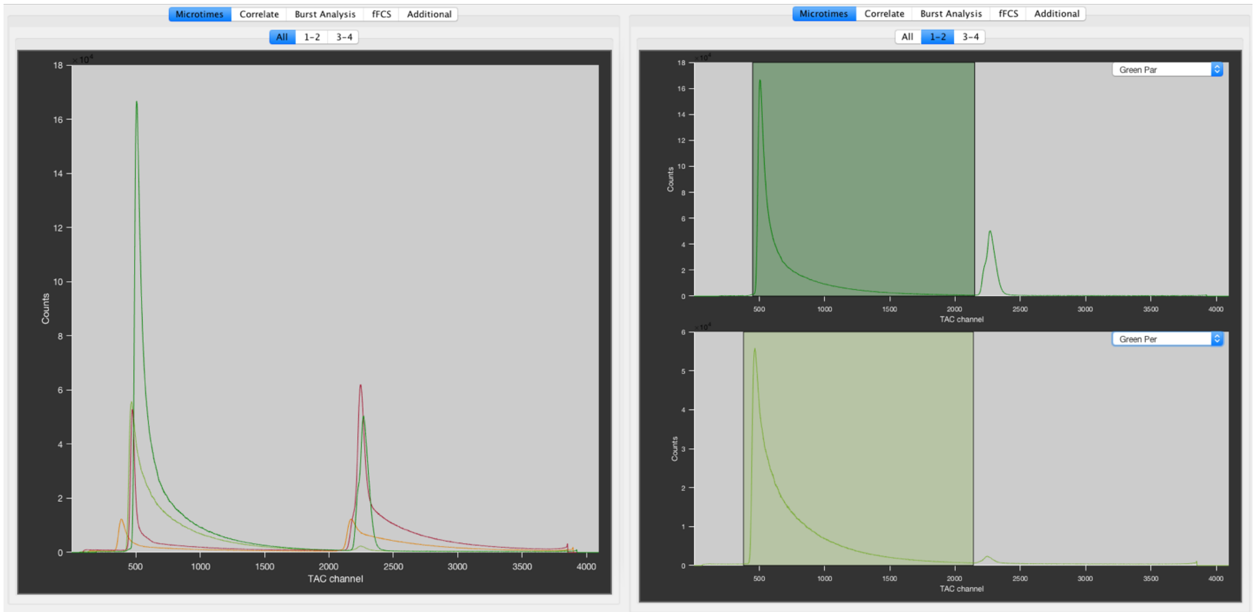


Fig. 3: The **All** microtime tab (left) and the *individual* microtime tab (right). The settings are 2 tabs and 2 plots per tab. The first tab *1-2* shows the two green detection channels, the second tab *3-4* shows the two red detection channels. The green shaded areas indicate the defined PIE channels on the green detection channels.

The microtime histograms of the defined detection channels are plotted on the right. The **All** tab show all microtime histograms in a single plot.

The individual plots can be customized using the settings in the **Settings** tab in the bottom left of **PAM**. Choose the number of tabs and the number of plots per tab, then define the detection channel for each plot using the dropdown menu in the top left of the respective plot.

PIE channels will be shown as shaded areas in the respective PIE channel color. You can directly click the individual microtime plots to define the microtime range of the PIE channels as described in *Defining PIE channels*.

The x-axis unit for the microtime histogram can be changed between TAC channels and time in nanoseconds using the checkbox in the **Settings** tab.

Right-clicking the plots allows switching between log and lin scales, and allows showing/hiding the scatter and *IRF* data.

Intensity Trace

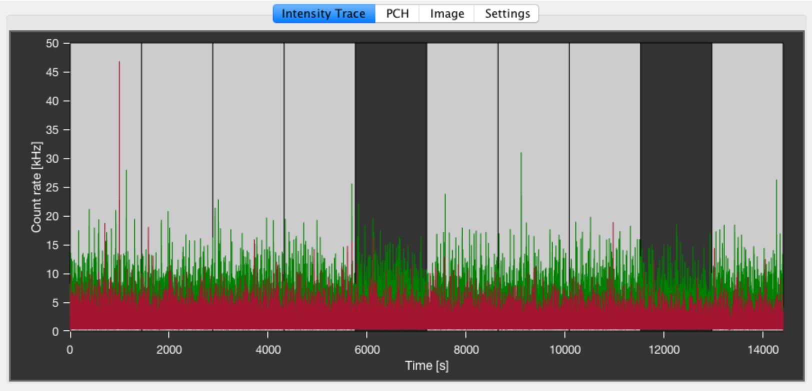


Fig. 4: Example time trace of two PIE channels with unselected macrotime blocks (dark blocks).

The **Intensity Trace** tab shows the intensity time trace for the selected PIE channel(s). The binning of the signal trace can be changed in the **Settings** tab.

For FCS analysis, the measurement can be split up into several blocks of equal size. To select/unselect a certain time range, *left-click* on the respective block, turning it dark. This will exclude it from FCS analysis. A second *left-click* will re-enable the block. In the **Settings** tab, one can change how the blocks are defined using the **Macrotime sectioning type** dropdown menu. *Constant time* will split the measurement using a constant time interval per block, specified by the value for the **Sectioning time** in seconds. *Constant number* will split the measurement into a number of blocks specified by the **Section number**, adjust the time per block based on the measurement duration.

Photon counting histogram

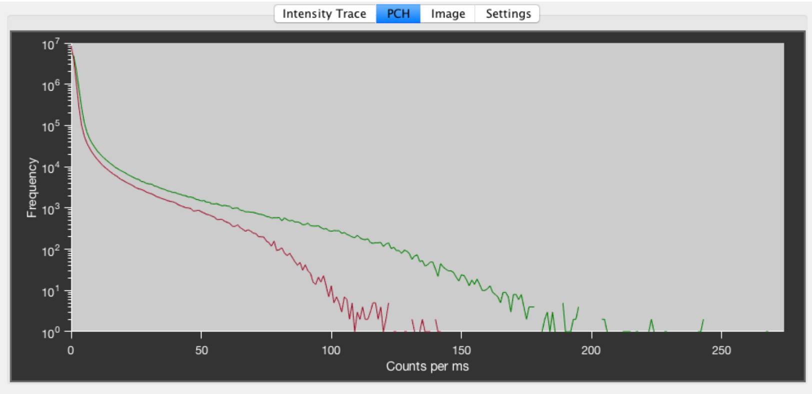


Fig. 5: Example photon counting histogram of a dilute solution. The peak at low photon counts is background noise, while the tail towards higher photon counts originates from the fluorescent molecules.

The **PCH** tab show a histogram of the photon counts per millisecond for the selected PIE channel(s). It can be useful to identify bright particles like aggregates in the solution. Fitting of photon counting histograms as described in [Chen1999] is currently not implemented in **PAM**.

Image

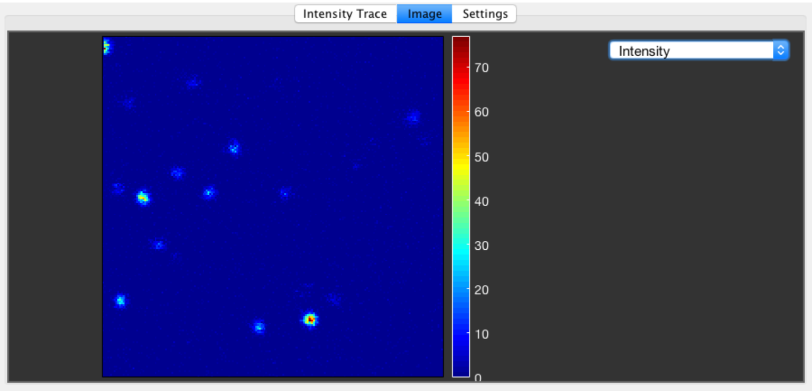


Fig. 6: The **Image** tab.

If the data has the required information to be processed into an image (i.e. pixel/line/frame markers), a raw image of the selected PIE channel can be shown in the **Image** tab. In addition to the *intensity image*, a *lifetime image* based on the mean microtime can be calculated by checking the respective checkbox in the **Settings** tab. Switch between *intensity* and *lifetime image* using the dropdown menu in the **Image** tab.

Hint: The scaling and colormap of the image can be adjusted with the matlab colormap editor by right clicking the colorbar. Click on the image once before adjusting to make sure that the image is the active axes.

2.1.5 Exporting raw data

The data of each PIE channel can also be exported, either to the matlab main workspace or saved as a standard file format (e.g. TIFF or Text). To do that, select the PIE channels to be exported in the **PIE channel list** and right click to and choose the *Export…* menu. There are different exporting options available:

…Raw data (total): Copies the raw photon data of the PIE channels to the main workspace.

…Raw data (per file): Copies the raw photon data of the PIE channels to the main workspace, separated by the individual files loaded.

…image (total): Opens a new figure displaying a summed up images and copies the intensity data to the main workspace.

…image (per frame): Copies the intensity data per frame to the main workspace as 3D arrays.

…image (as .tif): Saves the 3D image stack as a TIFF in the selected folder with the name of the first file and the PIE channel as appendix.

…microtime pattern (as .dec): Save the microtime histogram data as a *.dec* file for loading into the lifetime analysis sub-program. The *.dec* file are simply text files.

A more advanced method of exporting PIE channel based images is described in *Batch analysis*.

2.1.6 IRF and background/scatter

Per PIE channel, the ‘instrument response function’ (IRF) pattern and the scatter pattern are needed to perform reconvolution lifetime analyses in *TauFit*, *Burst Analysis*, the average background count rate is needed to *correct burst data*.

Sharp data such as the IRF is additionally needed to carry out *Detector calibration*.

Recording the IRF:

- Some dyes with a long lifetime can be quenched to an IRF by using a saturated solution of potassium iodide (KI), for example ATTO488 or ATTO655. Always prepare the KI solution fresh, or store it aliquoted in the freezer. Avoid KI crystals by spinning down the tube and pipetting only the supernatant onto the coverslip. Add concentrated dye to the KI solution to achieve the needed signal intensity at the laser power normally used. If necessary, increase or decrease the laser power. Focus far enough from the coverslip to be sure to record at constant intensity.
- Scattering of water (only works if the emission filter transmits a bit of excitation light). Reflection of laser light on a glass coverslip is not a good idea, as the glass might contribute to background signal with a lifetime.
- Removing the emission filter and recording laser reflections. Be careful not to damage your detectors!!!
- A dye with an intrinsically short lifetime (malachite green, bengal roze), although these typically look a bit different than the actual IRF.

Before storing the IRF pattern, consider using it to check possible *count rate dependent shifting of the used detector(s)*.

Recording the scatter pattern / background: Record a measurement of the buffer, accounting for any impurities that might be present in the sample.

Storing the IRF pattern: To store the IRF pattern for all PIE channels in the Profile from a single *IRF* measurement, load the measurement into **PAM** and select *File -> Store loaded data as IRF (for all PIE channels)*. This will save the microtime pattern of the currently loaded data for all *PIE channels* as the associated IRF pattern. To store the IRF for individual PIE channels, instead *right-click* the PIE channel that you want to set the IRF for and select *Save IRF for the selected PIE channel*.

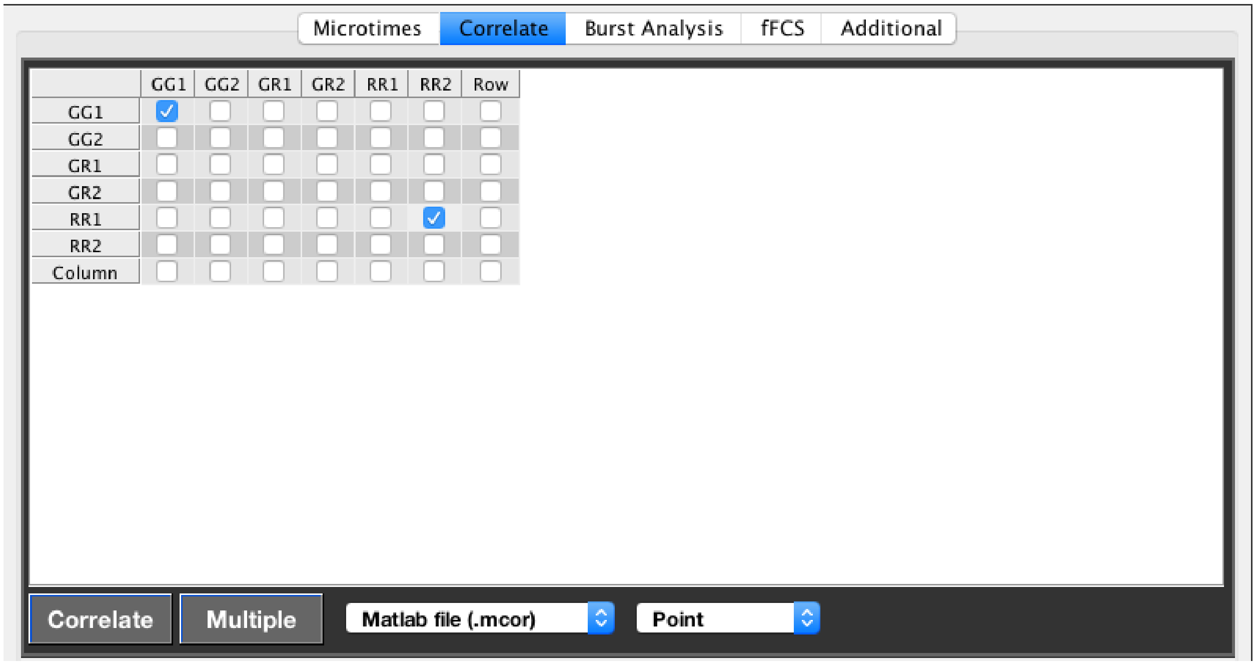
Storing the scatter pattern / background: Load the measurement and select *File -> Store loaded data as Scatter/Background*. This will save the microtime pattern of the currently loaded data for all *PIE channels* as the associated background/scatter pattern and store the count rates of the individual PIE channels as the background count rates.

You can display the currently stored *IRF* and background patterns in the microtime plots by *right-clicking* the microtime plot and selecting *Plot IRF* or *Plot Scatter Pattern*.

2.1.7 Using parallel processing

Parallel processing using multiple cores of the CPU based on MATLAB **parfor** loops is currently implemented for the computation of correlation functions, 2CDE filter calculation, burstwise lifetime fitting and simulation of photon data. When using this feature, consider that it takes some time to set up the parallel pool (~1 minute). You can enable this feature using the menu *Extras -> Parallel Processing -> Enable Multicore*. The number of cores can be defined using the *Number of cores* option. Consider using less cores than physically available if you want to perform other tasks while the analysis runs.

2.2 Fluorescence Correlation Spectroscopy



	GG1	GG2	GR1	GR2	RR1	RR2	Row
GG1	☑	☐	☐	☐	☐	☐	☐
GG2	☐	☐	☐	☐	☐	☐	☐
GR1	☐	☐	☐	☐	☐	☐	☐
GR2	☐	☐	☐	☐	☐	☐	☐
RR1	☐	☐	☐	☐	☐	☑	☐
RR2	☐	☐	☐	☐	☐	☐	☐
Column	☐	☐	☐	☐	☐	☐	☐

Fig. 7: The current selection will calculate the autocorrelation of channel *GG1* and the cross-correlation between the channels *RR1* and *RR2*.

Calculation of fluorescence correlation functions is performed on the basis of the defined PIE channels using the correlation grid in the GUI. Autocorrelation functions fall on the diagonal line, while cross-correlation functions are performed by checking off-diagonal elements. Usage of the last row or column will select all elements in the respective row/column. Clicking the bottom right checkbox will select all auto-correlations.

Correlation functions are calculated in one direction in time, delaying the column channel $I_2(t+\tau)$ with respect to the row channel $I_1(t)$:

$$G(\tau) = \frac{\langle I_1(t) I_2(t+\tau) \rangle}{\langle I_1(t) \rangle \langle I_2(t) \rangle}$$

Here, $<>$ denotes time averaging over the measurement. Temporal auto- and cross-correlation curves are calculate based on a multiple-tau correlation algorithm (see [Felekyan2005] for details). The time axis follows a multiple-tau scheme, doubling the time interval every 20 steps in units of the smallest available time difference given by the macrotime clock of the measurement. (The time delay axis is thus given by [1,2,3. . . .,18,19,20,22,24,. . . 36,38,40,44,48,. . .] and so forth.)

Clicking **Correlate** will perform the chosen correlation for the currently loaded file. Fitting of the correlation functions is performed in *FCSFit* or *Pair correlation analysis*, depending on the data type.

Multiple files can be correlated in one step by clicking the **Multiple** button. Select multiple files to be correlated using the open-file dialog. All files will be treated as currently defined.

Correlation functions are generally saved in a MATLAB file format (*.mcor*), allowing the data to be easily loaded into the MATLAB workspace again. Alternatively, the correlation results can be saved in a general text file (*.cor*) for processing in other software. Both save formats can also be chosen simultaneously. Choose the output format using the left popup menu.

The user can choose to coarse-grain the time axis by applying a **Divider**, available via right-click on the correlation grid. This divider effectively reduces the maximum achievable time resolution by the specified factor. If, for example, a clock tick is 100 ns and a divider of 10 is specified, the earliest time bin will span 1 μs. This divider might be useful when the laser repetition does not match the macrotime clock (e.g. laser only pulses every *n* th clock tick).

PAM currently implements five correlation methods which can be chosen using the right popup menu.

2.2.1 Point-FCS

Choose **Point** if you only want to calculate the temporal correlation function of the photon trace. Individual correlation functions are determined for each macrotime block (see *intensity trace plot*), which are used to address the error in the correlation function by calculating the standard error of the mean as:

$$SEM = \frac{\sigma}{\sqrt{N}}$$

where σ is the standard deviation and *N* is the number of macrotime blocks.

It may be of interest to exclude certain time intervals from the calculation of the correlation function (e.g. if large aggregates traversed the focus). This can be achieved before the calculation by unchecking the respective macrotime block, or after the calculation by clicking the respective correlation function in the preview plot. Selected curves are shown as solid lines, while unselected curves are plotted as dashed lines.

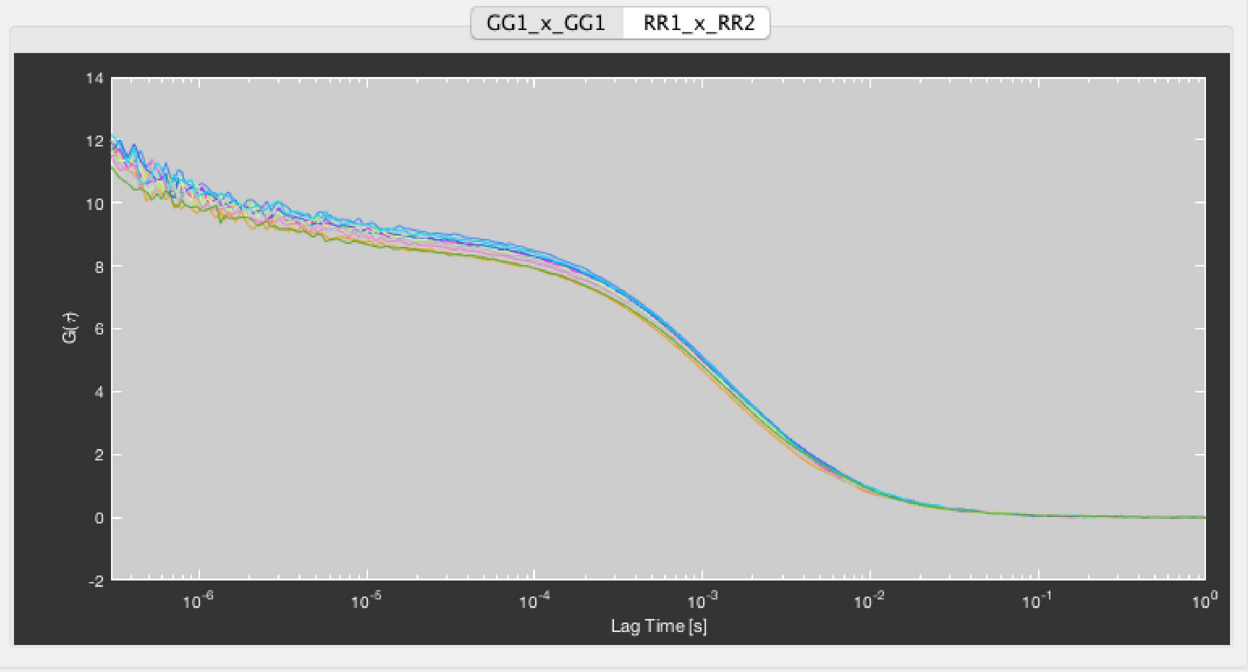


Fig. 8: The correlation preview window.

By default, all calculated curves are saved in the results file, so no further action is required if no curve is to be excluded. If a selection is made, confirm it by *right-clicking* the plot and choosing **Save selected correlations**. The right-click menu additionally allows to select or unselect all curves at once.

Suppression of afterpulsing

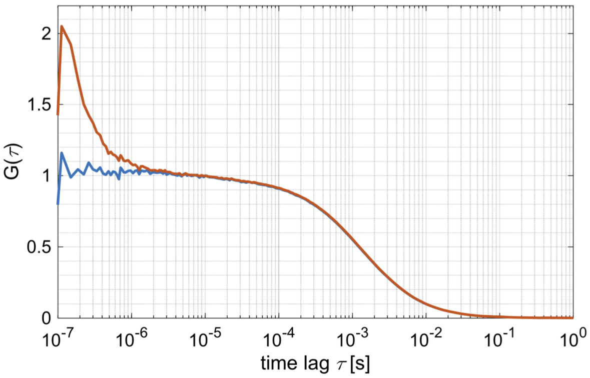
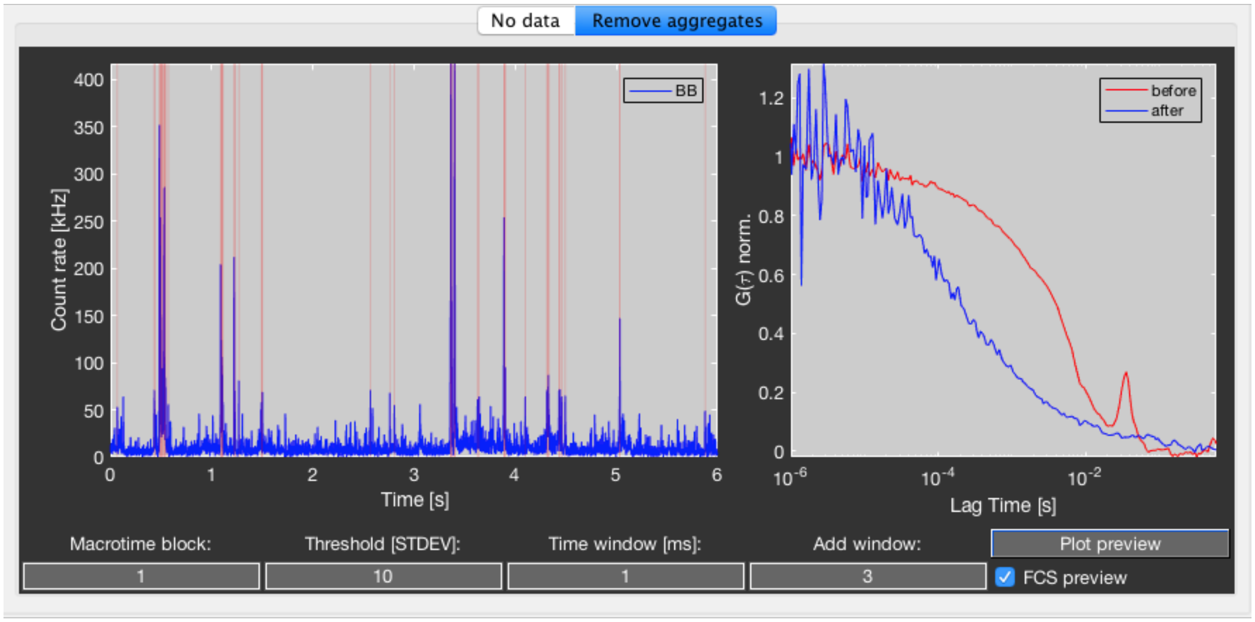


Fig. 9: Removal of afterpulsing by fluorescence lifetime correlation spectroscopy. Red: no correction, blue: corrected.

For autocorrelation analysis, detector afterpulsing can be filtered by selecting the **Correct for afterpulsing** checkbox. This will apply a flat background correction using FLCS to suppress the effect of afterpulsing of signal recorded on a single detector. See [Enderlein2005] for details.

Removal of aggregates



Macrotime block: 1

Threshold [STDEV]: 10

Time window [ms]: 1

Add window: 3

☑ FCS preview

FCS measurements are prone to artifacts cause by large aggregates. Since the contribution of species scales with the square of the brightness, single large aggregates can dominate the correlation function.

Use the **Remove aggregates** tab to preview the parameters for the removal of aggregates. The algorithm will perform a burst search by using a sliding time window, which will identify signal spikes (see *here* for more details). The count rate threshold that has to be exceeded is specified in units of the signal standard deviation. Tune the size of the time window depending on the diffusion time of the aggregates that should be excluded. Additionally, you can add a time window around the edges of detected spikes to further stretch out the regions where aggregates are detected. The algorithm will erase the spike region and fill them with random Poissonian noise.

Choose the macrotime block (based on the *sectioning of the measurement*) you want to preview and click the **Plot preview** button. This will display the signal trace binned using the specified time window and highlight excluded time regions by **red** shading. Check the **FCS Preview** checkbox to also get a preview of the uncorrected FCS curve (red) and the corrected FCS curve (blue). If multiple autocorrelations are selected in the correlation table, only the first selection is displayed in the preview module.

Check the **Remove aggregates** checkbox under the correlation table to enable filtering using the specified criteria for the FCS calculation.

Note: Afterpulsing correction and aggregate removal are exclusive and can not be combined.

2.2.2 Nanosecond-FCS

Nanosecond-FCS (or full-FCS) uses the microtime information in addition to the macrotime information to reconstruct the photon arrival time with highest time resolution (given by the resolution of the photon microtime), by:

$$t_i = \mathrm{MT}_i \cdot \mathrm{TAC} + \mathrm{MI}_i$$

Here, TAC is the TAC range, MT is the macrotime and MI is the microtime.

Two modes are available for nanosecond-FCS. **Microtime** will perform the correlation using a multiple-tau time axis as used for point-FCS, while **Microtime (linear)** will instead use a linear time axis from -10 μs to 10 μs with a resolution of 100 ps, based on the interphoton time distribution as described in [Nettels2007].

2.2.3 Pair correlation function

Fluorescence pair correlation analyzes the temporal fluorescence correlation spatially resolved for scanning data. If a certain pattern was repeatedly scanned with a constant frequency, the data will show a direct relationship between the measurement time and position. **PAM** uses this relationship to divide the photons by position, by separating them into (temporally) equally sized bins, defined by the **#Bins** parameter and the scanning frequency (*FileInfo.ScanFreq*).

> **Note:** For continuous scanner movement, the number of bins should be chosen such, that the focus movement during the bin duration is significantly smaller than the focus size, generally <50nm. If scanning was discontinuous (i.e. the focus jumps between positions and stays there), the number of bins should be equal to the number of scanned positions, or a multiple of that, if the distance between positions is very small.

Each bin now only contains photons that originates from a particular focus position. Since the data now have temporal gaps, the timing for the correlation algorithm is reduced to the scanning frequency. Thus, the temporal resolution of the correlation functions is determined by the scanning frequency and not the sampling speed.

For each spatial bin, a temporal auto-correlation is calculated. Additionally, temporal cross-correlations between different bins and for different spatial lags are calculated. In most cases, not all spatial lags are needed for analysis, e.g. for very large distances, the correlation amplitudes are too low to be properly analyzed. For such cases, the user can limit the spatial lag range to be calculated, by using the **Distance** field and standard matlab vector indexing (e.g. 1:50 or [9,10,12], etc.), thus reducing the calculation time. The individual calculations are performed with the same algorithm as normal FCS.

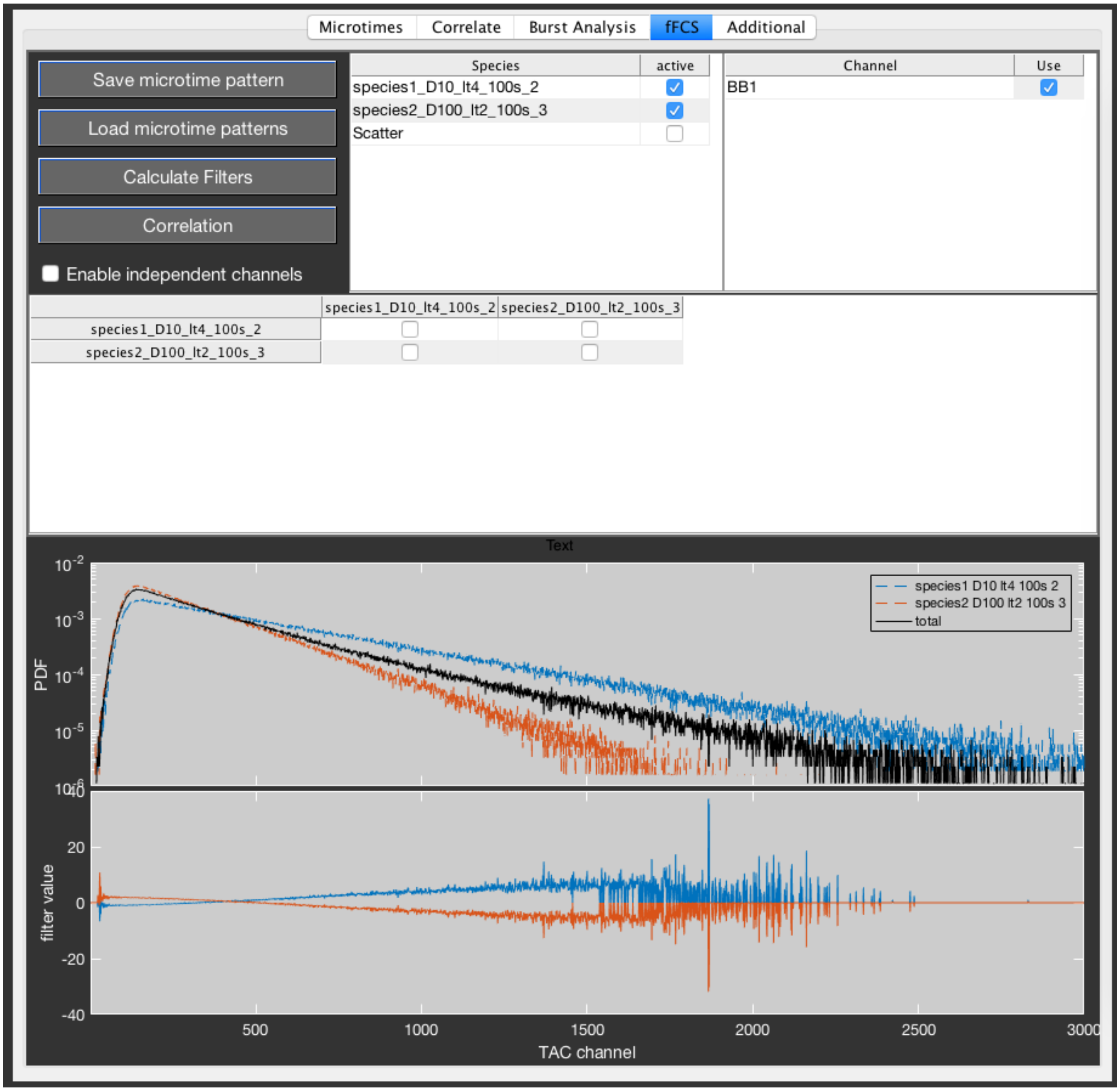
Note: The terms used in the case of pair correlation refer to the *spatial* auto- and cross correlations. But pair correlation also works for correlating signals from different channels (e.g. color or polarization). That means, it is possible to calculate spatial *and* spectral cross-correlations. The same selection of PIE channels as in standard FCS is used.

The pair correlation data is save a matlab bases *.pcor* format and can be easily loaded into the matlab main workspace or analysed with *Pair correlation analysis*.

2.2.4 FLCS/fFCS

Fluorescence lifetime correlation spectroscopy and filtered-FCS use differences in fluorescence lifetime or anisotropy to separate species in a mixture by assigning statistical weights to photon based on the microtime ([Boehmer2002] and [Felekyan2012]).

The basic assumption of FLCS/fFCS is that the microtime pattern of the measured mixture can be expressed as a linear combination of the microtime signatures of the individual species. One can define statistical filters from the linear decomposition which are used to assign weights to the individual photons based on their microtime.



☐ Enable independent channels

Species	active
species1_D10_lt4_100s_2	☑
species2_D100_lt2_100s_3	☑
Scatter	☐

Channel	Use
BB1	☑

	species1_D10_lt4_100s_2	species2_D100_lt2_100s_3
species1_D10_lt4_100s_2	☐	☐
species2_D100_lt2_100s_3	☐	☐

Fig. 10: The FLCS/fFCS GUI.

To perform fFCS analysis, the microtime patterns of the pure species need to be known. The current implementation assumes that a pure measurement of the components is available, or that they can be *extracted from a single-molecule measurement*. If the pure components are available, load the individual measurements and save the respective microtime pattern using the **Save microtime pattern** button. This will save all the microtime information in a *.mi* file.

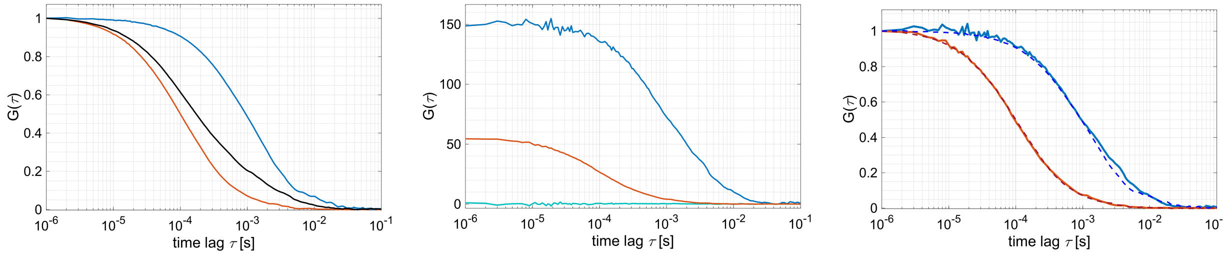


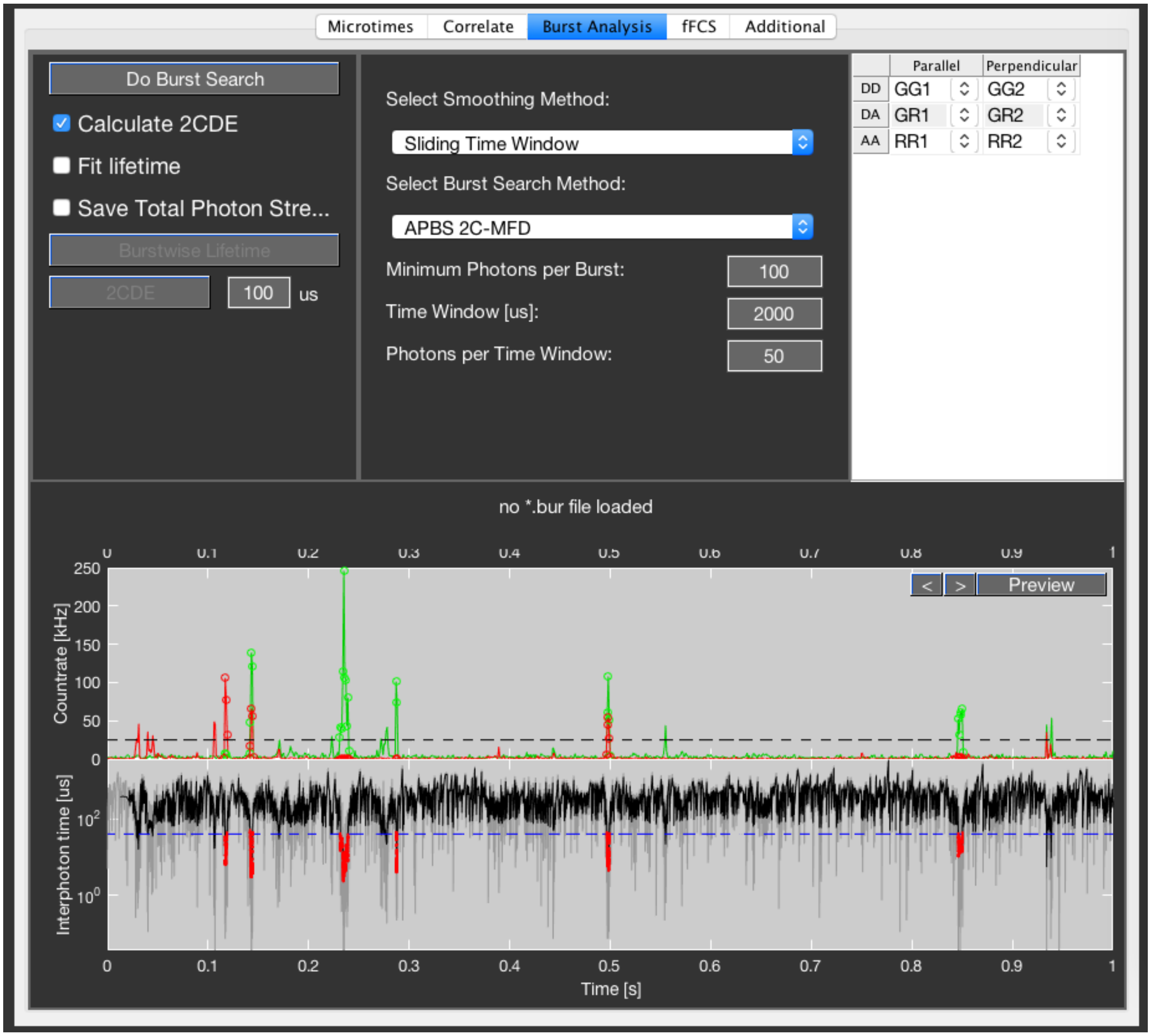
Fig. 11: Example FLCS analysis of two species with different diffusion coefficient and fluorescence lifetime (2 ns and 4 ns). (Left) Pure autocorrelation functions of fast diffusing species (red, lifetime of 2 ns), slow diffusing species (blue, lifetime of 4 ns) and a 2:1 mixture of fast and slow species. (Middle) Filtered species auto- and cross-correlation functions using the filters shown in the previous figure. (Right) Comparison of filtered auto-correlation functions from the mixture (solid lines) and the pure autocorrelation functions of the individual species (dashed lines).

Load the individual microtime patterns (*.mi* files) into **PAM** using the **Load microtime patterns** button. This will add the respective species to the *Species* table. Select the species to consider in the *Species* list. A *Scatter* species is always available based on the stored scatter pattern. *PIE channel* that are defined in all loaded *.mi* files will be listed in the *Channel* list. Select which *PIE channels* to use for filter generation. If multiple *PIE channels* are selected, the microtime histograms will be stacked for the global filter generation. Press the *Calculate Filters* button to generate the filters for the selected species.

Use the *Correlation* table to select which auto- and crosscorrelations are to be performed and press the *Correlation* button to perform the correlation.

Check the *Enable independent channels* checkbox to use two independent channels that can be crosscorrelated. This will allow to assign *PIE channels* to two independent channels. Filters will be independently generated for both channels and used in the crosscorrelation between the channels. Use this option to circumvent detector dead time and afterpulsing artifacts at sub-microsecond timescales by crosscorrelating signal detected on independent detectors.

2.3 Burst Analysis



☑ Calculate 2CDE

□ Fit lifetime

□ Save Total Photon Stre...

100 us

Select Smoothing Method: Sliding Time Window

Select Burst Search Method: APBS 2C-MFD

Minimum Photons per Burst: 100

Time Window [us]: 2000

Photons per Time Window: 50

	Parallel	Perpendicular
DD	GG1	GG2
DA	GR1	GR2
AA	RR1	RR2

Fig. 12: BurstSearch tab overview.

The **Burst Analysis** tab is used to extract single-molecule events from a PIE-MFD measurement. Different burst search methods are available, whose parameters should be adjusted for the specific measurement at hand. **PAM** calculates basic burst wise parameters like FRET efficiency and labeling stoichiometry automatically, while additional parameters (fluorescence lifetime, ALEX 2CDE filter. . .) can be calculated on demand.

2.3.1 Burst search methods and burst selection

Burst search methods are categorized by experiment type (two color, three color, polarized detection used or not) and by burst selection method (APBS, all photon burst search, or DCBS, dual channel burst search, [Nir2006]). Two color or three color experiments are denoted by *2C* or *3C*, respectively. The use of polarized detection is signified by appending the tag *MFD* or *noMFD*.

Burst selection

The general burst selection method is the *all photon burst search (APBS)*, which disregards the channel information and combines all photons into a single signal trace on which the burst search is performed. This way, single and double labeled molecules are detected likewise. This is the general setting that should be applied to unknown systems.

Dual channel burst search (DCBS) performs *APBS* burst searches on the signal trace after donor excitation (*GG* + *GR*) and after acceptor excitation (*RR*) separately. Only time intervals during which the two separate burst searches identified a burst are kept. This has some consequences for how different events are recorded. Firstly, donor and acceptor only events will not be detected. Secondly, bursts which show photobleaching of either donor or acceptor fluorophore will be truncated. Thirdly, blinking events of either donor or acceptor fluorophore will segment the single molecule event. *DCBS* is effective in limiting the analysis to only double labeled molecules.

Tip: Always start analyzing data with an *APBS*, to get an idea about the system and to look for possible artifacts (impurities, high amount of photobleaching) and to determine correction factors from the donor and acceptor only subpopulations (see BurstBrowser). Use *DCBS* when the system at hand is characterized and a quick analysis of the relevant double labeled molecules is desired.

Warning: When using *DCBS* only, be cautious of low FRET populations! If no double labeled molecules are present in solution, *DCBS* will still find multi-molecule events, which will have a low FRET efficiency. Look at the *APBS* to ensure that the detected low FRET species is indeed a population and not just a bridge between acceptor and donor only populations.

Smoothing and burst search methods

There are currently two different methods to smooth the signal trace and to perform burst thresholding.

Sliding time window: This method uses a sliding time window centered around each photon to estimate the local count rate. If the local count rate exceeds the threshold, the center photon is assigned to belong to a burst. If a certain number of consecutive photons pass the threshold, they are grouped into a burst. See [Nir2006] for more details. The user can specify the time window width in microseconds, the required photons per time window, and the minimum number of photons per burst. The time window *T* is symmetrically centered, i.e. ranging from *t-T/2* to *t+T/2*.

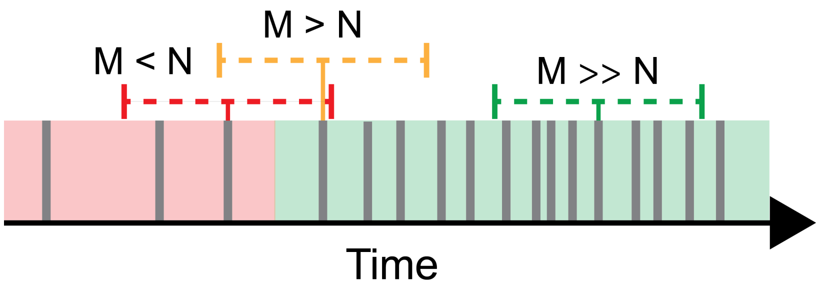


Fig. 13: Illustration of the sliding time window burst search. *N* is the threshold number of photons per time window and *M* is the actual number of photons in the illustrated time window. Individual photon arrival times are indicated by grey bars.

Interphoton Time with Lee Filter: This method selects bursts based on a threshold on the interphoton time, which will be short during a burst of photons compared to long interphoton times during background signal.The interphoton time trace is smoothed using a Lee filter, which has the property to smooth regions of constant signal, while leaving regions with rapid parameter changes (i.e. the edges of bursts) untouched. The user can specify the minimum number of photons in a burst, the size of the smoothing window in one direction *N*, and the interphoton time threshold in microseconds (lower values correspond to higher count rates). The smoothing window parameter *N* corresponds to a total number *2N+1* photons considered for the smoothing, centered around the respective photon. The smoothing parameter σ_0, which characterizes the noise standard deviation, is estimated by the global standard deviation of the interphoton times. For more information, see [Schaffer1999] and [Enderlein1997].

Burst search preview

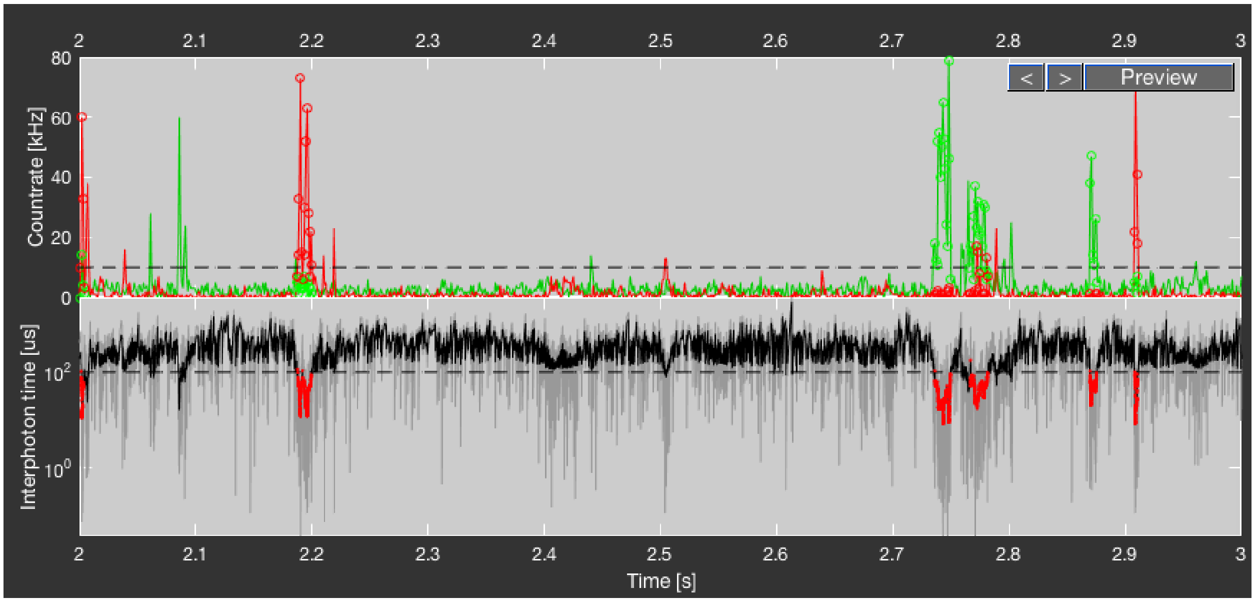


Fig. 14: The burst search preview window.

To judge the chosen parameters, a preview window is displayed in the bottom part of the **Burst Analysis** tab. Click the **Preview** button to update the preview display, and use the arrow buttons to go forward and backward in time by one second.

The top plot shows the intensity traces after donor excitation (*GG* + *GR*, green) and after acceptor excitation (*RR*, red) using a time bin of 1 ms. The dashed line shows the count rate threshold given by the burst search parameters. For *Sliding Time Window* it is given by the number of photons per time window divided by the time window length, whereas for *Interphoton Time with Lee Filter* it is given by the inverse of the interphoton time threshold. Time bins that belong to an identified burst are indicated by colored open circles.

The bottom plot shows the raw interphoton time trace in grey, together with the smoothed trace in black. The interphoton time threshold is shown by the dashed line. Photons that belong to a burst are colored red. If *Sliding Time Window* is selected, a default smoothing window of 30 is used for display, and the interphoton time threshold is given by the inverse of the count rate threshold.

Note: Depending on which burst search method is chosen, primarily use the respective plot (top for *Sliding Time Window*, bottom for *Interphoton Time*) to evaluate the burst search parameters. The respective other plot is just shown for illustrative purposes.

2.3.2 PIE channel assignment

The next step after choosing the experiment type and burst search method is to correctly assign PIE channel to FRET channels. If the standard PIE channel notation has been used (see *PIE channel notation*), this step is straight-forward. In the **Burst Analysis** interface, FRET channels are defined using the same notation as for PIE channels, with the more general notation of *D* for FRET donor and *A* for FRET acceptor. The FRET channels are thus *DD*, *DA*, and *AA*, each defined per detection polarization. Use the drop-down menu to select a PIE channel for each FRET channel. (For three-color FRET, the assignment table will instead denote channels by color as used for PIE channel definition, i.e. *BB*, *BG*, . . . etc.).

Note: It is not possible to change the photon assignment (i.e. the properties of the PIE channels) after the burst analysis has been performed.

Tip: Take the time to again check the detector assignment and microtime boundaries of the individual PIE channels. Common errors are the wrong assignment of parallel and perpendicular channels, and wrong microtime ranges that lead to false assignment of photons (e.g. because the timing of the hardware was changed).

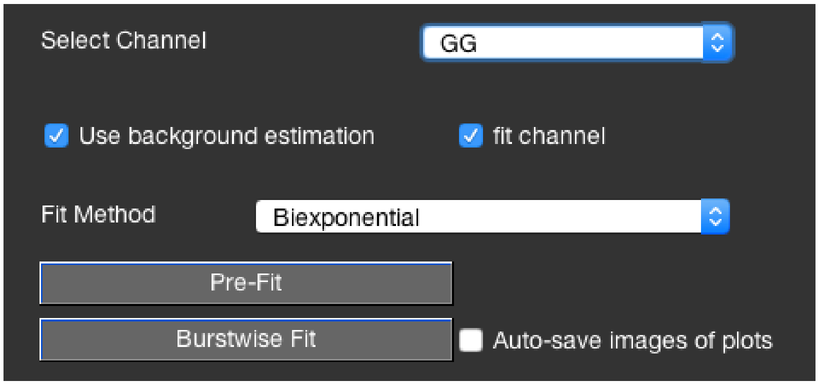
2.3.3 Performing the burst search

Once all parameters have been set, start the burst analysis by clicking **Do Burst Search**. Once the burst analysis is finished, it will create a subfolder in the location of the raw data with the name given by the raw data file. The burst data is stored in a file with the *.bur* ending. The name of the file is given by the file name of the raw data with appended burst search method. Raw photon data is saved in a separate file with ending *.bps* (for 'burst photon stream') to avoid having to reload the original data. Additionally, the specified burst search parameters as well as all other available meta data (see *Meta data saving*) is saved in a *.txt* file with the same name.

2.3.4 Additional burst wise parameters

After the burst search is performed, two more parameters can be determined: the fluorescence lifetime and the two-channel kernel density estimate filter. After the burst search, the respective buttons labels are red to indicate that the parameters have not been determined. After calculation the button labels turn green.

Burstwise fluorescence lifetime



Select Channel: GG

☑ Use background estimation ☑ fit channel

Fit Method: Biexponential

□ Auto-save images of plots

Fig. 15: Changed settings windows when calling **TauFit** for burstwise lifetime determination.

Click **Burstwise Lifetime** to open an instance of *TauFit*. The program structure is the same as described in the **TauFit** documentation. The main plot shows the cumulative microtime histograms of all selected bursts. Align the microtime channels as described in the **TauFit** documentation. The quality of the alignment can be judged using the **Pre-Fit** button, using a single or biexponential fit model. Use the popup menu to go through all colors (*GG* and *RR* for two-color experiments, *BB* additionally for three-color) to align all color channels and ensure the right correction factors are used. Click **Burstwise Fit** to start the lifetime fit.

Note: The donor decay usually shows multiexponential behavior (donor only molecules and one or more FRET species). Use the biexponential fit model to judge the correctness of the microtime shift parameters (most crucial is the IRF shift). The acceptor decay usually shows a monoexponential decay, so a single exponential is sufficient.

Uncheck the **fit channel** checkbox if you do not want to fit the lifetime for this channel. The **Auto-save images of plots** automatically saves pictures of the performed alignment and pre-fits so you can inspect it later to ensure the lifetime has been determined correctly.

The lifetime fitting is performed using a single exponential model function and is based on a maximum likelihood estimator as described in [Maus2001]. If the **Use background estimation** checkbox is enabled, a scatter contribution using the microtime pattern of the specified scatter measurements (i.e. of the used buffer) is added with a fixed fraction f_{BG}, estimated based on the total signal in the burst *S*, the background count rate *BG* and the burst duration *T*:

$$f_{\mathrm{BG}} = \frac{T \cdot BG}{S}$$

Two-channel kernel density estimate filter

Additionally, the two-channel kernel density estimate filter (2CDE) as described in [Tomov2012] can be calculated by clicking the **2CDE** button. The calculation is based on a single variable parameter, which is the time window to estimate the photon density. This parameter should be larger than the average interphoton time, but smaller than the burst length. Generally, a value of 100 μs yields good results. 2CDE filter calculation supports the usage of *multiple cores*.

Correlation analysis with time window (purified-FCS)

If you want to perform a correlation analysis on a subset of bursts with the option of adding a time window around the burst to correctly determine the diffusion time of molecules, it is required to check the **Save total photon stream** checkbox. It is currently not possible to re-access raw data to perform this step at a later time point, so when correlation analysis is to be performed, the decision has to be made at the time point of the analysis. The total photon stream is saved as an *.aps* file. See *Burstwise FCS* for more information on using a time window in burstwise FCS.

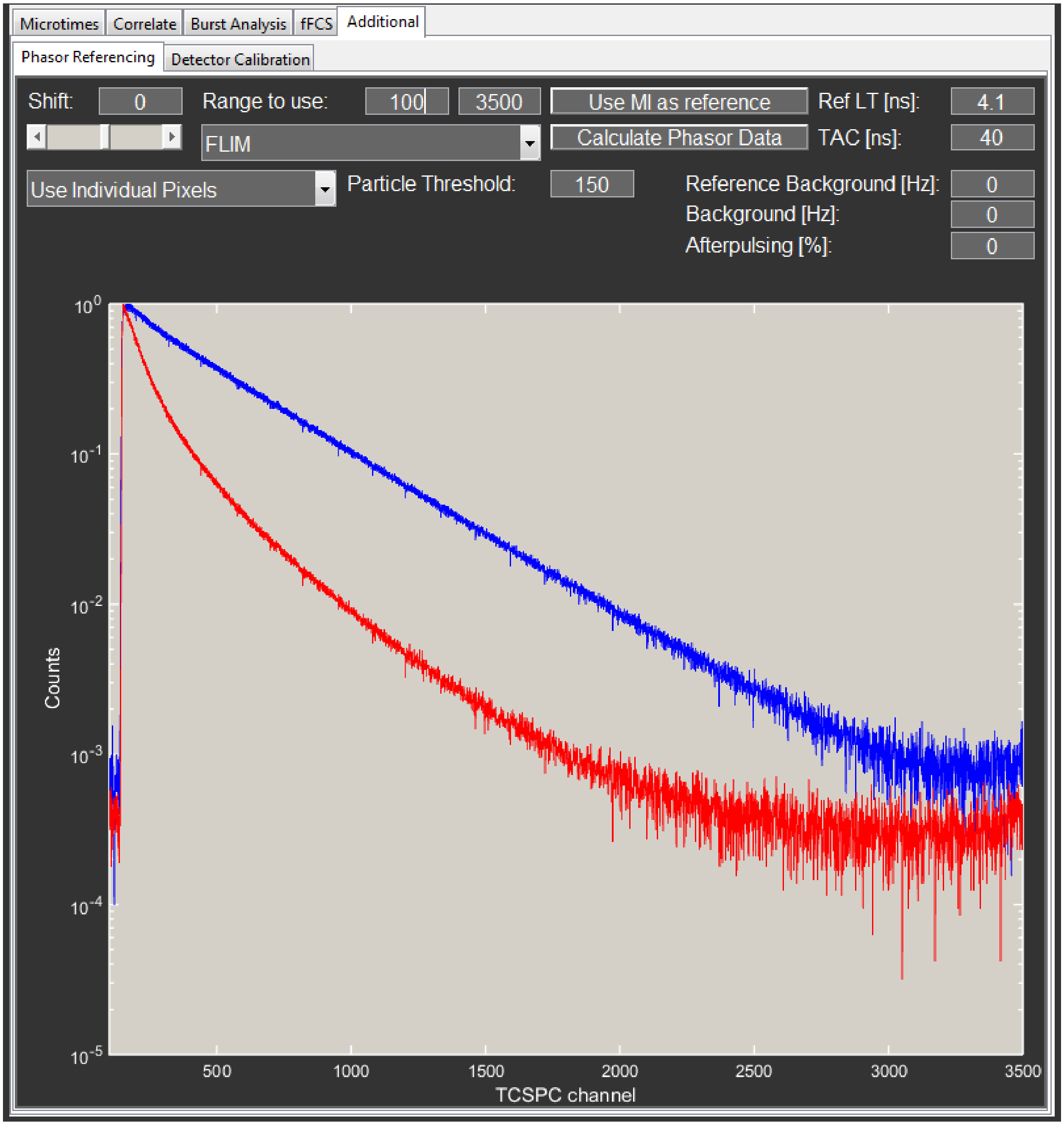
Note: The resulting *.aps* file will be of similar size as the raw photon file. Be aware of this when disk space is a concern.

Using a total measurement for photon distribution analysis

Photon distribution analysis (PDA) can be performed on the total photon stream. To export the total measurement to **PDAFit**, *right-click* the **Do Burst Search** button and select *Export total measurement to PDA*. This will chop up the data into time bins of 1 ms. Currently, this option is only available for two-color MFD data.

2.4 Phasor referencing

The **Phasor Referencing** tab is used to convert the raw photon data into referenced phasor FLIM data to be analyzed with *Phasor Analysis* sub-program. The processed data is saved as a *.phr* file that is based on the standard *.mat* matlab data format and can be easily loaded into the matlab command window.



Shift: 0

Range to use: 100 3500

Ref LT [ns]: 4.1

TAC [ns]: 40

FLIM

Use Individual Pixels

Particle Threshold: 150

Reference Background [Hz]: 0

Background [Hz]: 0

Afterpulsing [%]: 0

2.4.1 Reference data and settings

To account for the IRF of the system, a reference file of known mono-exponential lifetime is needed (an *IRF* measurement with a lifetime of 0 also works) for all calculations. During referencing, the program calculates the raw phasor of the reference and compares it to the expected values (based on the reference lifetime set with the corresponding editbox). The parameters (phase and modulation) needed to move the measured to the expected values represent the influence of the instrument and are then applied to the actual measurement to remove the bias from the IRF.

A phasor plot depends on the underlying frequency. **PAM** uses the full range of the microtime histogram as the period of this frequency.

The reference histogram is stored in the *UserValues* information of the current profile with the **Use MI as reference** button and is displayed in blue in the phasor histogram plot.

Note: The phasor referencing works on the full detector channel and not just the PIE-channel. The current detector channel can be selected with the corresponding popup-menu.

2.4.2 Background corrections

Uncorrelated stray light and detector dark-counts result in an additional (infinite) lifetime component for the phasor calculations. Since the extent of the background influence depends on the total signal, it has to be corrected individually for each pixel. In **PAM** the user can set a background count rate for both the reference and the measurement files. Additionally, an afterpulsing probability can be set to further correct the signal.

Note: In most cases afterpulsing affects the reference and the measurement data to the same extent and thus is removed automatically in the referencing procedure. This is however no longer the case if the used range is restricted. In practice, for high enough signals (SNR>100) no corrections are necessary.

Note: Background originating from the sample, most commonly cell auto-fluorescence, has usually an unknown lifetime and cannot be removed during the referencing. Its influence has to be considered during data interpretation.

2.4.3 IRF shift and artifacts

Some detectors (mostly APDs) used for FLIM applications show shifts of the *IRF* based on the signal intensity and other environmental effects. In **PAM**, fast, count-rate dependent influences can be removed during data loading (see *Detector calibration*). To correct for very slow changes (on timescales longer than a measurement), the user can shift the reference relative to the measurement histogram with the corresponding editbox or slider.

Certain regions of the microtime range can contain artifacts that strongly influence the phasor calculations. A common sources for such artifacts is spectral cross-talk in multi-color PIE experiments. If these artifacts are sufficiently separated in time from the actual signal, they can be simply removed by limiting the **Range to use**. To limit the effect of this cutting, the remaining range should be at least a factor of 5 longer than the longes relevant lifetime component (usually >25ns per color or repetition rates <40MHz).

Note: The phasor frequency is no affected by the used range, bins outside of the range are simply set to 0. Thus, experiments with different ranges (e.g. singe and multi-color experiments) can be compared, as long as artifacts from cutting can be neglected and the microtime ranges are the same.

2.4.4 Phasor calculation

Once all relevant parameters are set, the phasor data for the currently loaded data can be calculated and saved by clicking on the **Calculate Phasor Data**. Hereby, all frames and files are added, resulting in a single phasor frame. Most of the calculation is performed using C++ to decrease the calculation time.

2.4.5 Particle detection

Usually, a phasor is calculated for each pixel individually. However, in some cases it might be useful to pool several pixels for the calculation to increase the signal-to-noise or decrease measurement time. This is especially useful when an average phasor of individual well-separated particles or cells is desired. For such cases, a *Particle detection* module is currently in development. To generate the correct input type for this, switch from *Use Single Frame* to *Framewise Phasor* and calculate the phasor as usual. This saves the data as a normal, summed up *.phr* file for standard phasor analysis and also as a *.phf* file that contains the information for each frame and pixel and can be loaded with the *Particle detection*) module.

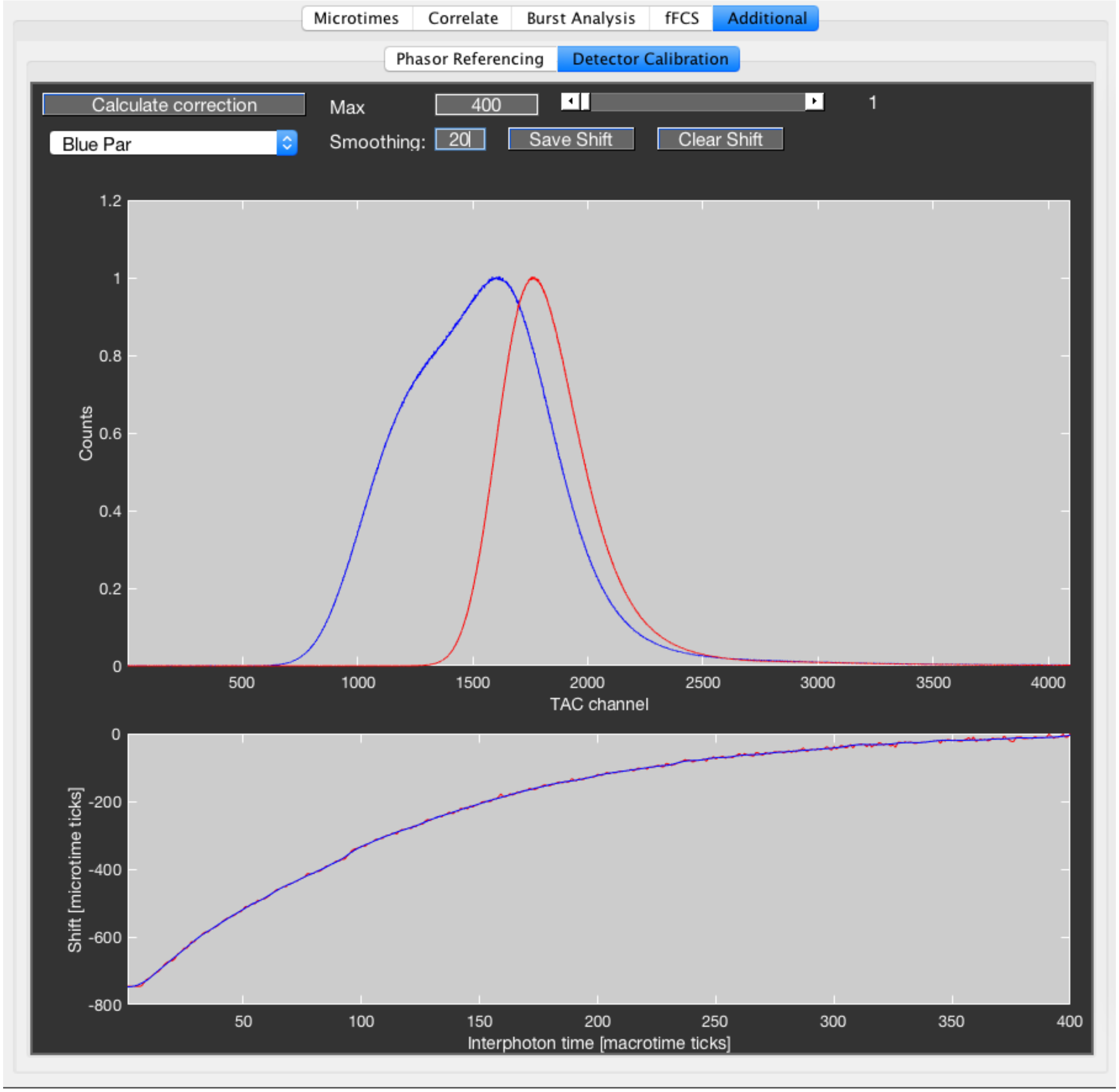
2.5 Detector calibration

Avalanche photodiodes used for single-photon counting often show a count rate dependent shift of the instrument response function. In our experience, the COUNT modules from LaserComponents are affected more strongly than the commonly used SPCM modules from Perkin Elmer (see [Otosu2013]). However, if the effect is accounted for, COUNT detectors have a superior IRF. The degree of this effect varies on a module-by-module basis, and has thus to be calibrated for every detector individually. For Perkin Elmer APDs (and maybe others) some other artifacts occur that cannot be easily corrected for.

Before attempting to correct data, first verify whether your detector shows a count rate dependent IRF shift. We find it simplest to either record images with highly varying count rates (e.g. a mixture of dimly and brightly fluorescent cells), or to record a time trace where the laser power is modulated between high and low. If no obvious artifacts appear in the experimental decay, the detector likely does not require any correction.

To be able to clearly identify a possible correlation between the photon arrival (micro-)time and the (macro-)time between photons, and to sample long, but also short interphoton times enough, it's best to avoid a large spread on the photon arrival time. Therefore, only use high intensity *IRF* data (~100 kHz) recorded for 5-10 min to have >30 million photons.

To compute the calibration curve, photons are sorted based on the macrotime difference to the previous photon, and microtime histograms are calculated for the sampled interphoton times. The peak positions of the microtime histograms are extracted to calculate the IRF shift with respect to the IRF at long interphoton times.



Max: 400

Smoothing: 20

Fig. 16: The detector calibration GUI.

To determine the detector calibration, load a measurement and select the detection channel using the dropdown menu. Select the range of interphoton times in units of the macrotime clock that should be calculated (the range should extend to at least 20 μs). Click **Calculate correction** to compute the calibration.

The top plot shows the uncorrected (blue) and corrected (red) microtime histogram. Using the slider in the top right, one can examine the individual uncorrected microtime histograms for the selected interphoton time (plotted in green). The bottom plot shows the calculated correction lookup table in red. If a smoothing larger 1 is selected using the edit box, the smoothed correction curve is shown in blue. Smoothing is performed using a simple moving average filter of specified size.

Save the shift using the **Save Shift** button. If a previous calibration is stored, clear it first using the **Clear Shift** button. The correction is applied on read-in of the raw data, so the current data has to be reloaded for the correction to be applied.

You can export the current detector calibration using *File -> Load/Save Calibrations... -> Save Detector Shifts to File* to a *.sh* file. Reload stored detector shifts using *Load Detector Shifts*. This will overwrite currently stored detector calibrations in the profile.

Note: The calibration is defined based on units of the macrotime clock and specific to the used TCPSC resolution. If any setup parameters change, the calibration has to be redone.

2.6 Batch analysis

Many of the routine data processing in **PAM** can be automated and performed sequentially for multiple datasets without further user input. This automated processes can be started from the **Database** tab in the top left of **PAM**. The list on the left shows all the files that are to be processed. To add files to the database, go to the menubar ans select *File -> Database...*. The *Add individual TCSPC files to database* option adds each selected file as an individual entry. The *Add connected TCSPC files to database* option will tread the selected files as a single entry that is then processed together (e.g. when individual frames are saved in separate files). The later option continues prompting the user with a file selection window until *cancel* is pressed. The actual data is only loaded once the actual processing is activated.

2.6.1 Image exporting

To export multiple image stacks at once, press the **Export TIFFs** button. An individual TIFF stack is created for each PIE channel (selected in the top right of the **Database** tab) and each selected entry in the *Database list*. Additionally, the user can subdivide these into different groups (e.g. for z-stacks) with the editboxes next to the button. The first editbox determines the number of different stack (e.g. different z-positions) and the second number corresponds to the number of consecutive stacks. For example, 5 and 1 will create 5 individual stacks and each stack contains every fifth frame. 2 and 5, on the other hand, will frames 1-5 into the first stack, 6-10 into the second, 11-15 again into the first, and so on. The stacks are saved into the folder of the raw data with the filename of the first file and an appendix for the PIE channel and the stack position.

2.6.2 Microtime histogram

To export multiple microtime histograms at once, press the **Export Microtime Histogram** button. An individual *.dec* file is created for each PIE channel (selected in the top right of the **Database** tab) and each selected entry in the *Database list*. The files are saved into the folder of the raw data with the filename of the file and an appendix for the stored PIE channels. *.dec* files are text-based and can be loaded into **TauFit** for analysis at a later time point.

2.6.3 Correlate

Correlation for all selected entries in the **Database list** can be performed by pressing the **Correlate** button in the **Database** tab. This process uses the settings in the **Correlate** tab and the segmentation from the **Settings** tab.

2.6.4 Burst analysis

Burst analysis for all selected entries in the **Database list** can be performed by pressing the **Burst analysis** button in the **Database** tab. This process uses the settings in the **Burst Analysis** tab.

Hint: In general it is best to first perform burst analysis for a single file to optimize the settings and then perform the batch analysis with the rest of the measurements.

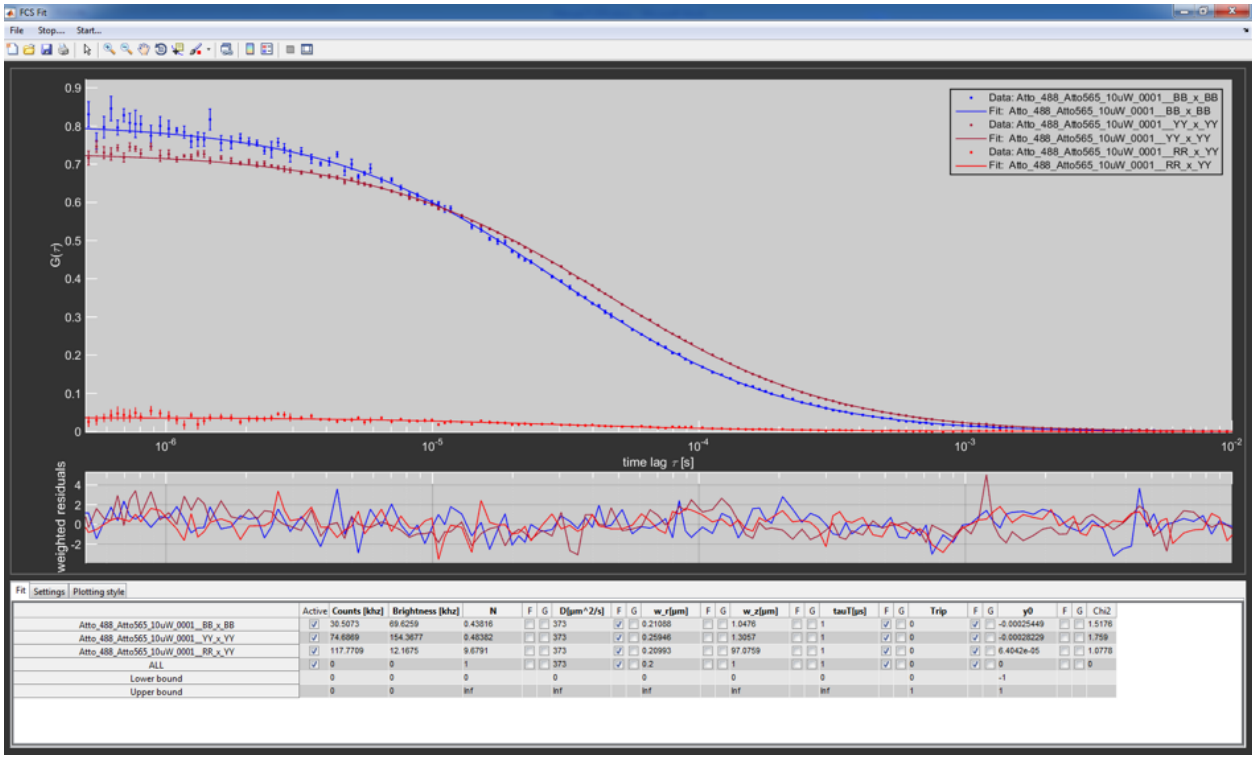
CHAPTER 3

FCSFit

Contents

Fitting of FCS data can be accessed from the PAM main window by clicking on **Advanced Analysis -> FCSFit** or from the command window by typing in `FCSFit`.

The upper part of the window is dedicated for displaying the FCS data, fits and corresponding residuals. In the lower part of the window the user can set the fit parameters and other fitting and display settings.


FCS Fit
File
Stop...
Start...
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
G(τ)
Data: Atto_488_Atto565_10uW_0001__BB_x_BB
Fit: Atto_488_Atto565_10uW_0001__BB_x_BB
Data: Atto_488_Atto565_10uW_0001__YY_x_YY
Fit: Atto_488_Atto565_10uW_0001__YY_x_YY
Data: Atto_488_Atto565_10uW_0001__RR_x_YY
Fit: Atto_488_Atto565_10uW_0001__RR_x_YY
10^{-6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}
time lag τ [s]
weighted residuals
4
2
0
-2
Fit
Settings
Plotting style

	Active	Counts [khz]	Brightness [khz]	N	F	G	D[μm^2/s]	F	G	w_r[μm]	F	G	w_z[μm]	F	G	tauT[μs]	F	G	Trip	F	G	y0	F	G	Chi2
Atto_488_Atto565_10uW_0001__BB_x_BB	☑	30.5073	69.6259	0.43816	□	□	373	☑	□	0.21088	□	□	1.0476	□	□	1	☑	□	0	☑	□	-0.00025449	□	□	1.5176
Atto_488_Atto565_10uW_0001__YY_x_YY	☑	74.6869	154.3677	0.48382	□	□	373	☑	□	0.25946	□	□	1.3057	□	□	1	☑	□	0	☑	□	-0.00028229	□	□	1.759
Atto_488_Atto565_10uW_0001__RR_x_YY	☑	117.7709	12.1675	9.6791	□	□	373	☑	□	0.20993	□	□	97.0759	□	□	1	☑	□	0	☑	□	6.4042e-05	□	□	1.0778
ALL	☑	0	0	1	□	□	373	☑	□	0.2	□	□	1	□	□	1	☑	□	0	☑	□	0	□	□	0
Lower bound		0	0	0			0			0			0			0			0			-1			
Upper bound		0	0	Inf			Inf			Inf			Inf			Inf			1			1			

Fig. 1: FCSFit main gui

3.1 Data handling and loading

Currently, the program only supports files created with PAM, both based on MATLAB files (*.mcor*) and text files (*.cor*).

To load new files, got to **File -> Load New Files** or **File -> Add Files**. The former clears previously loaded files and load the new ones, while the later adds new files to the loaded ones. All loaded files are listed in the table and the bottom. Individual files can be removed by clicking on the **Remove** checkbox in the **Plotting Style** tab.

The raw correlation data is accessible from the command line via the global variable *FCSData*, and the fits and other processed parameters are stored in the global variable *FCSMeta*.

3.1.1 Saving the analysis state

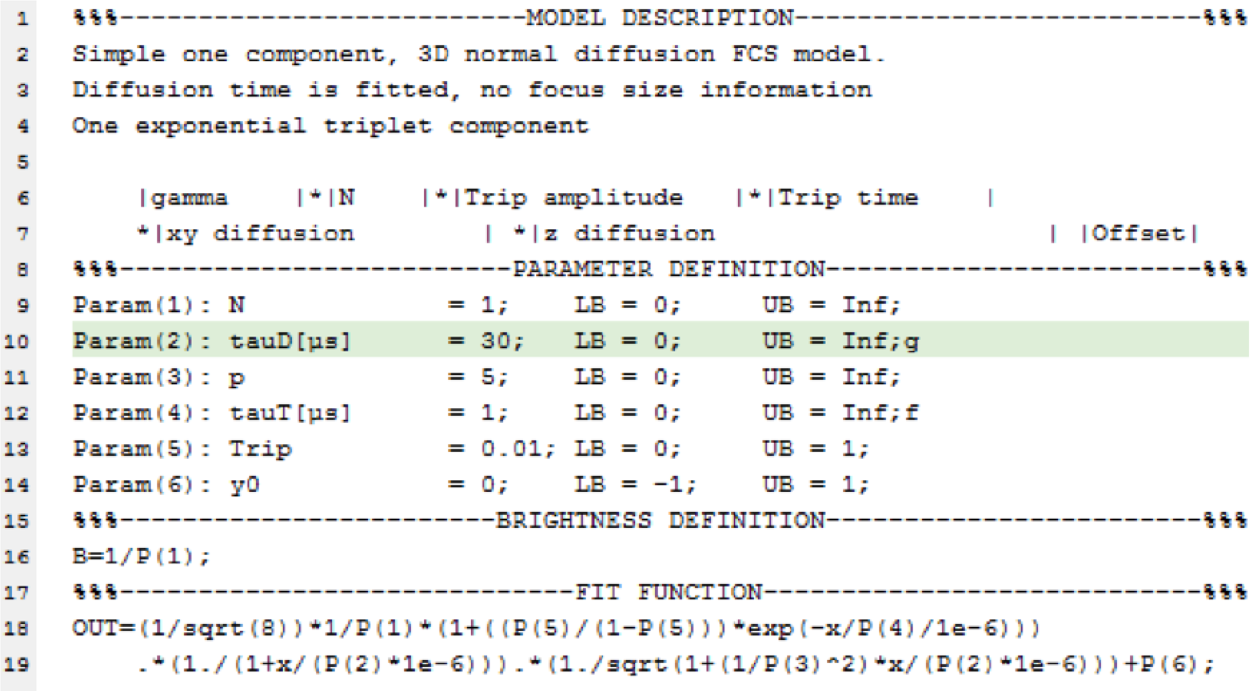
The analysis state can be saved as a session (*.fcs* file) under *File -> Save FCSFit Session*. This will save all fitted parameters including global and fixed flags, the loaded model function, all settings and the visualization state (line types, curve colors…). Load a previous session using *File -> Load FCSFit Session*.

3.2 Fitting models and algorithms

Fitting in **FCSFit** is not based on hard coded fit models but instead uses models stored in text files (standard *.txt*). This allows users to easily create new models and modify old ones without having to change the code itself, especially useful for a compiled program.

3.2.1 Fit models

FCSFit uses text based fit models. A simple example of the structure of a model is shown here:



```
%%%--------------------------MODEL DESCRIPTION--------------------------%%%
Simple one component, 3D normal diffusion FCS model.
Diffusion time is fitted, no focus size information
One exponential triplet component

    |gamma    |*|N    |*|Trip amplitude   |*|Trip time    |
    *|xy diffusion        | *|z diffusion                     | |Offset|
%%%-------------------------PARAMETER DEFINITION------------------------%%%
Param(1): N             = 1;    LB = 0;     UB = Inf;
Param(2): tauD[µs]       = 30;   LB = 0;     UB = Inf;g
Param(3): p             = 5;    LB = 0;     UB = Inf;
Param(4): tauT[µs]       = 1;    LB = 0;     UB = Inf;f
Param(5): Trip          = 0.01; LB = 0;     UB = 1;
Param(6): y0            = 0;    LB = -1;    UB = 1;
%%%------------------------BRIGHTNESS DEFINITION------------------------%%%
B=1/P(1);
%%%-----------------------------FIT FUNCTION----------------------------%%%
OUT=(1/sqrt(8))*1/P(1)*(1+((P(5)/(1-P(5)))*exp(-x/P(4)/1e-6)))
    .*(1./(1+x/(P(2)*1e-6))).*(1./sqrt(1+(1/P(3)^2)*x/(P(2)*1e-6)))+P(6);
```

Fig. 2: Example model text file

Each model file consists of four segments. The headers of the segments are needed for the program to recognize the individual parts and should not be modified.

Model description: This part is not used by the program itself but is used as a description for the user to understand the purpose and outline of the fit.

Parameter definition: In this section the different parameters used for the fit are defined and initial values set. The nth parameter is defined by *Param(n)*:, followed by the name of the parameter displayed in **FCSFit** (do not use spaces in the name as they are used as separators). The next settings in order are the initial value, the lower and the upper limits. At the end you can use *g* or *f* to the set the parameter to global or fixed during initialization of the fit. The equal signs and semicolons are essential as the program searches for them as separators between the entries.

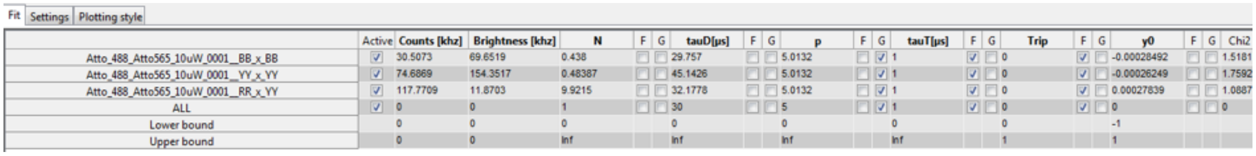
Brightness definition: The program uses this part to calculate the molecular brightness. Generally, the brightness is defined as *1/N*, but can be more complex for other models. Use standard MATLAB nomenclature to define this function and terminate it with a semicolon.

Fit function: Here the actual function used for fitting is defined. The individual parameters are abbreviated as *P(n)*. Use standard MATLAB nomenclature to define the function. Multiline entries are possible. In such a case use simple line breaks and not the `...` used in MATLAB scripts and functions.

3.2.2 Fitting procedure

When starting **FCSFit**, the model used during the last session is pre-loaded. A different fit function can be selected via **File -> Load Fit Function**. The standard folder for this is the **models** sub-folder of **PAM**.

The file names and the fit parameters defined in the model are shown in the table in the **Fit** tab.



	Active	Counts [khz]	Brightness [khz]	N	F	G	tauD[μs]	F	G	p	F	G	tauT[μs]	F	G	Trip	F	G	y0	F	G	Chi2
Atto_488_Atto565_10uW_0001__BB_x_BB	☑	30.5073	69.6519	0.438	□	□	29.757	□	□	5.0132	□	☑	1	☑	□	0	☑	□	-0.00028492	□	□	1.5181
Atto_488_Atto565_10uW_0001__YY_x_YY	☑	74.6869	154.3517	0.48387	□	□	45.1426	□	□	5.0132	□	☑	1	☑	□	0	☑	□	-0.00026249	□	□	1.7592
Atto_488_Atto565_10uW_0001__RR_x_YY	☑	117.7709	11.8703	9.9215	□	□	32.1778	□	□	5.0132	□	☑	1	☑	□	0	☑	□	0.00027839	□	□	1.0887
ALL	☑	0	0	1	□	□	30	□	□	5	□	☑	1	☑	□	0	☑	□	0	□	□	0
Lower bound		0	0	0			0			0			0			0			-1			
Upper bound		0	0	Inf			Inf			Inf			Inf			1			1			

Fig. 3: The fit table

The different rows correspond to the individual files. The last three rows are used to set parameters for all files, the lower and the upper bounds, in order.

The first column after the filenames is used to (de-)activate individual files to include them in or exclude them from the fitting and displaying procedure. The following column shows the average count rate of the file. For cross-correlation it is the sum of both channels. This is followed by the molecular brightness, calculated from the brightness defined in the model time the count rate. The other columns correspond to the different fit parameters with three columns each. The first is an editbox that contains the actual value. The other two are checkboxes used to either fix a parameter (F) or to globally link it between all files (G). The very last column represents the $\chi^2_{\text{red.}}$ of the fit.

For the actual fitting, **FCSFit** uses the standard MATLAB least-square curve fitting algorithm (*lsqcurvefit*). Only the data-points displayed in the upper plot (defined in the **Settings** tab) are used for fitting. When no variables are linked globally, each file is fit individually. The free parameters are used for the fit, while the fixed variables are given to the function as additional data to calculate the functions. For a global fit, the variables and fit-functions are concatenated and a single fit is executed. After the fit is finished, confidence intervals are calculated and the plots and tables updated.

3.3 Additional settings and exporting

With the **Plotting Style** tab the user can change the appearance of the plotted data and fits. The **Settings** tab contains contains a variety of parameters for fitting and displaying, most notably the time range to be displayed and used for fitting (**Borders**), whether or not to use weights (**Use weights**) and other fit related settings. Furthermore, it contains general setting for exporting the curves to a new MATLAB window.

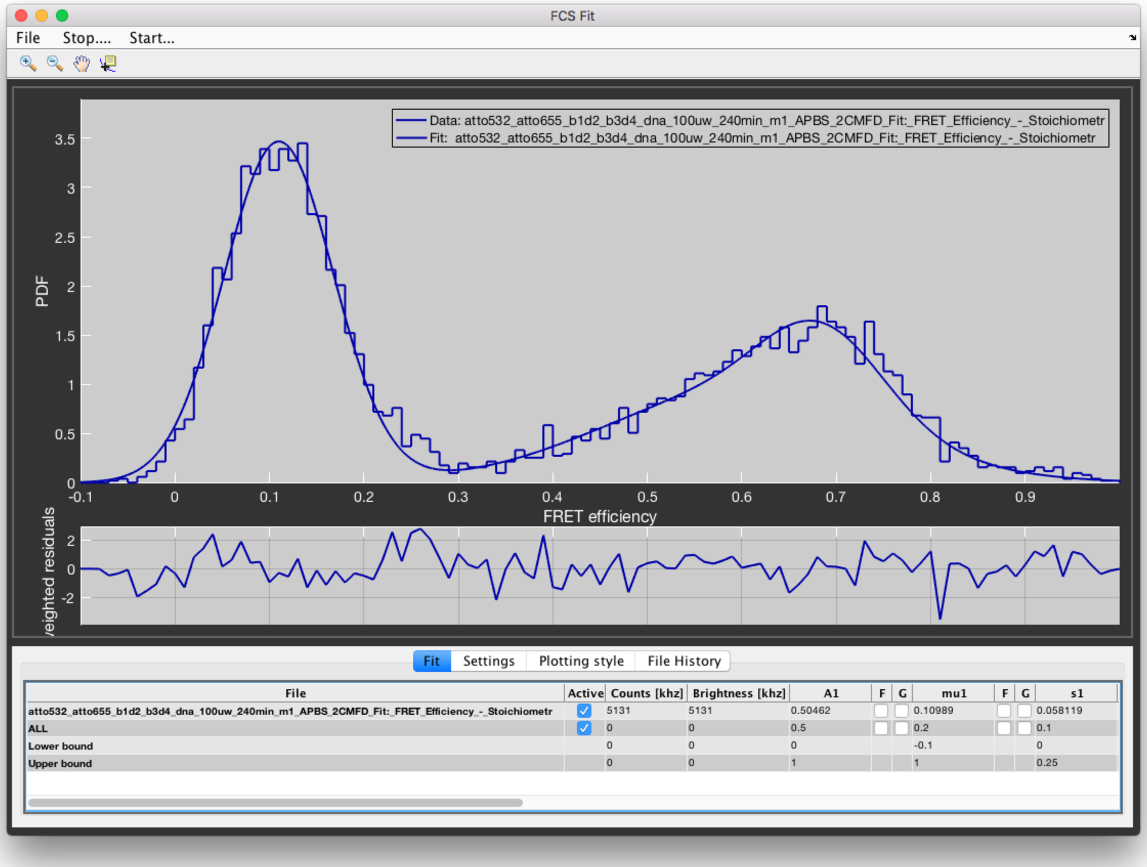
The curves can be exported to a new figure or the data can be exported to the MATLAB command line by right clicking the plot and selecting the appropriate function. Additionally, the **Copy Results to Clipboard** button copies the fit parameters to the clipboard so they can be parsed into standard programs like excel of origin.

3.4 Alternate mode for global fitting of FRET efficiency histograms

FCSFit can also be used to fit FRET efficiency histograms exported from **BurstBrowser** (see *Exporting FRET efficiency histograms*). Upon file loading, select "FRET efficiency data from BurstBrowser" to load *.his* files. Currently, models are available for fitting of normal and β-distributions. Note that the β-distribution is only defined in the interval [0,1].

If **Plot errorbars** is checked, the data is shown as line plot with error bars. Error bars are determined assuming Poissonian counting statistics for the histogram, i.e. by $\sigma = \sqrt{N}$ where N is the number of counts. If this option is unchecked, data will be shown as stair plots.

FRET efficiency data can be rebinned by changing the bin size (default bin size: 0.01). Choose an appropriate value depending on the sampling of the FRET efficiency histogram.


FCS Fit
File
Stop....
Start...
Data: atto532_atto655_b1d2_b3d4_dna_100uw_240min_m1_APBS_2CMFD_Fit:_FRET_Efficiency_-_Stoichiometr
Fit: atto532_atto655_b1d2_b3d4_dna_100uw_240min_m1_APBS_2CMFD_Fit:_FRET_Efficiency_-_Stoichiometr
PDF
3.5
3
2.5
2
1.5
1
0.5
0
-0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
FRET efficiency
weighted residuals
2
0
-2
Fit
Settings
Plotting style
File History

File	Active	Counts [khz]	Brightness [khz]	A1	F	G	mu1	F	G	s1
atto532_atto655_b1d2_b3d4_dna_100uw_240min_m1_APBS_2CMFD_Fit:_FRET_Efficiency_-_Stoichiometr	☑	5131	5131	0.50462	□	□	0.10989	□	□	0.058119
ALL	☑	0	0	0.5	□	□	0.2	□	□	0.1
Lower bound		0	0	0			-0.1			0
Upper bound		0	0	1			1			0.25

Fig. 4: FRET efficiency histogram analysis in FCSFit using Gaussian distributions

CHAPTER 4

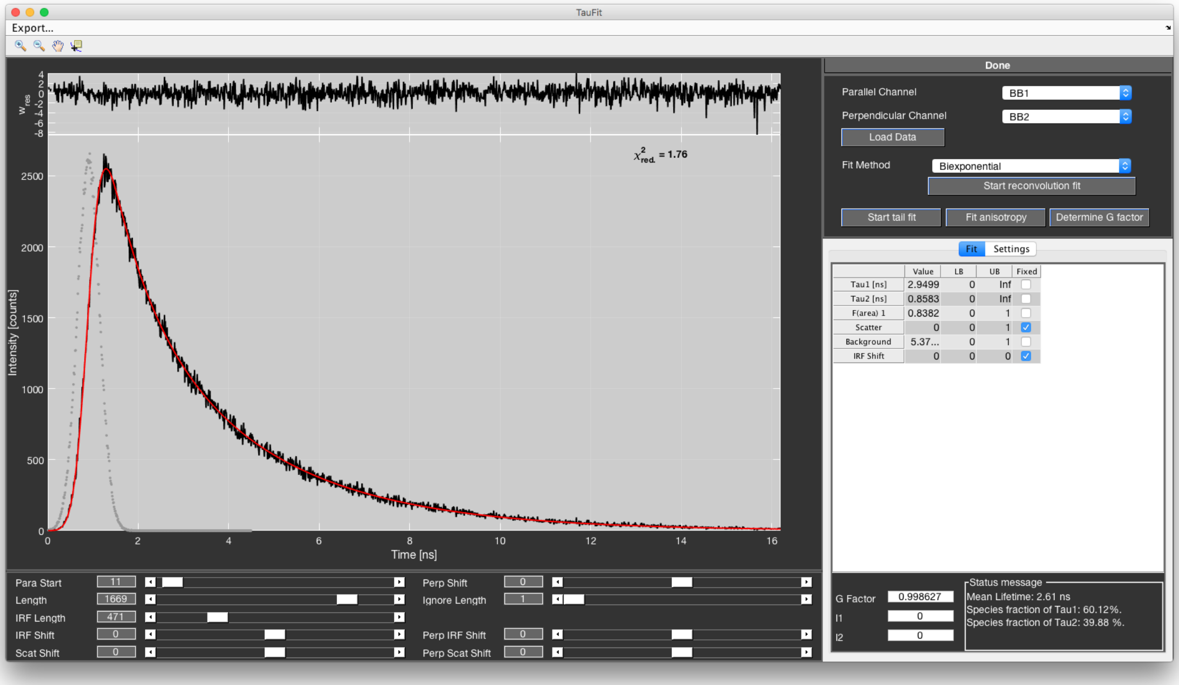
TauFit

Contents

TauFit is the central TCSPC module of PAM, dedicated to the analysis of fluorescence intensity decays and time-resolved anisotropy. The following analysis methods are available:

- Tail fitting of fluorescence decays and time-resolved anisotropy
- Reconvolution fitting of fluorescence decays using:
  - Single, bi- or triexponential model
  - Distribution of donor-acceptor distances
  - Time-resolved anisotropy model
- Simulation of κ^2-distributions from fundamental and residual anisotropies

TauFit integrates with both PAM for ensemble-TCSPC analysis and BurstBrowser for burst-selective sub-ensemble TCSPC analysis of species.



Parallel Channel: BB1

Perpendicular Channel: BB2

Fit Method: Biexponential

	Value	LB	UB	Fixed
Tau1 [ns]	2.9499	0	Inf	☐
Tau2 [ns]	0.8583	0	Inf	☐
F(area) 1	0.8382	0	1	☐
Scatter	0	0	1	☑
Background	5.37...	0	1	☐
IRF Shift	0	0	0	☑

Para Start: 11

Length: 1669

IRF Length: 471

IRF Shift: 0

Scat Shift: 0

Perp Shift: 0

Ignore Length: 1

Perp IRF Shift: 0

Perp Scat Shift: 0

G Factor: 0.998627

I1: 0

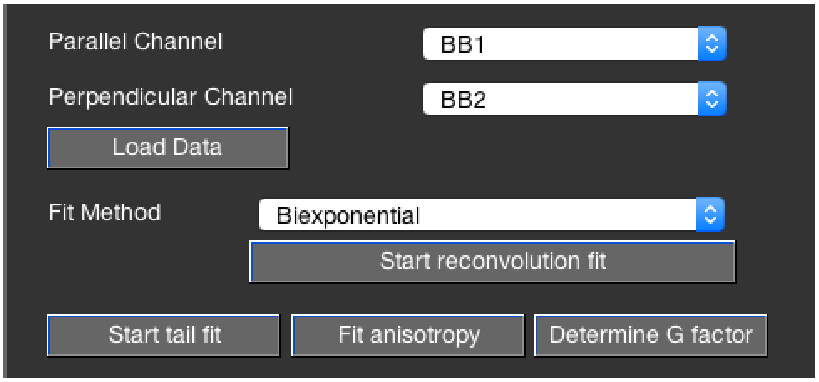
I2: 0

Status message
Mean Lifetime: 2.61 ns
Species fraction of Tau1: 60.12%.
Species fraction of Tau2: 39.88 %.

Fig. 1: TauFit's main GUI.

4.1 Data loading

4.1.1 From PAM (ensemble-TCSPC)



Parallel Channel: BB1

Perpendicular Channel: BB2

Fit Method: Biexponential

Fig. 2: The selection GUI.

TauFit directly accesses the data loaded into **PAM** via the defined PIE channels. To load fluorescence decay data into **TauFit** from **PAM**, choose the PIE channels for the parallel and perpendicular fluorescence signal using the popupmenus and click **Load Data**. If no polarized detection is used, simply select the same channel twice.

4.1.2 From BurstBrowser (subensemble-TCSPC)

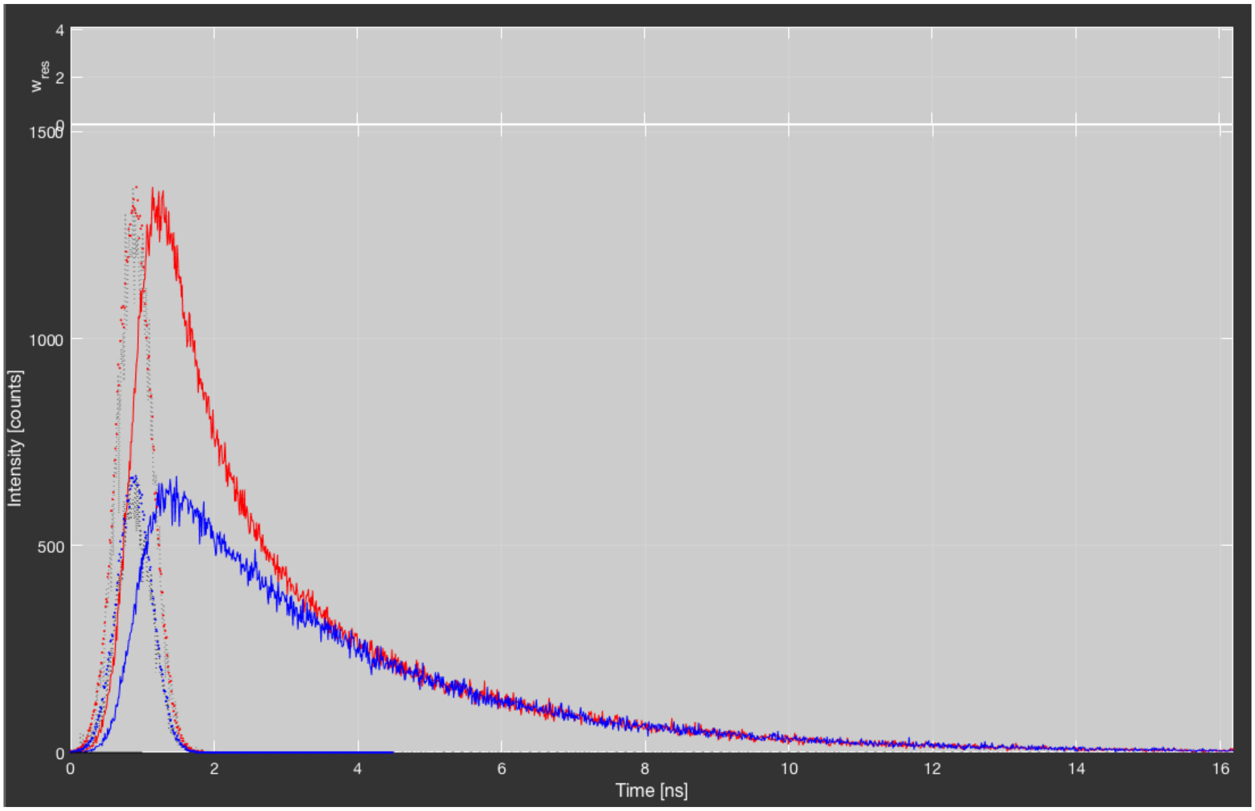
To analyze the fluorescence decay of a species from a burst measurement, you can directly export it from BurstBrowser as described in the respective section in the *BurstBrowser manual*.

4.1.3 From intermediate file format .decay

PAM allows to export the microtime decays of a raw data file into a processed *.dec* file as described in the respective section of the *PAM manual*. To load *.dec* files, select *File -> Load decay data (.dec)*. Note that this will put **TauFit** into *external data* mode, disabling interaction with the currently loaded data in either **PAM** or **BurstBrowser**. Close **TauFit** and open a new instance to re-enable the default mode.

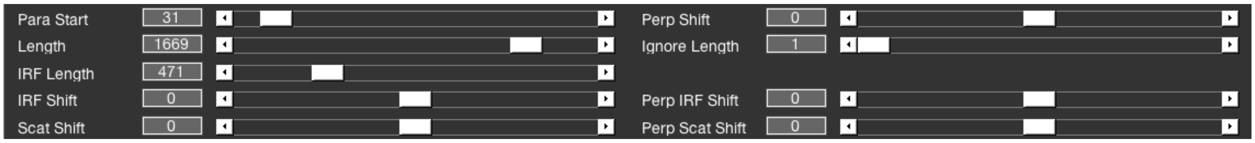
4.2 Aligning the data

4.2.1 Main plot



The main plot shows the raw fluorescence decay data of the selected PIE channels, using red for the parallel channel and blue for the perpendicular channel. The respective instrument response functions (*IRF*) are plotted as dotted lines. Additionally, the scatter patterns for both channels are indicated as dotted gray lines. If the ignore slider is used, a vertical black line indicates the start point of the data that is consider for the goodness-of-fit calculation.

4.2.2 Alignment sliders



Para Start 31
Length 1669
IRF Length 471
IRF Shift 0
Scat Shift 0
Perp Shift 0
Ignore Length 1
Perp IRF Shift 0
Perp Scat Shift 0

Fig. 3: The alignment sliders.

An important step when fitting fluorescence decays obtained on avalanche photodiodes is the alignment of the channels with respect to each other. The sliders should be used in the proposed order:

- **Start:** Global start of the data range with respect to the PIE channel range, i.e. a value of 10 omits the first 10 TCSPC bins.

- **Length:** Total length of the data range.
- **Perp Shift:** Shift of the perpendicular decay with respect to the parallel decay. Affects both the data and the IRF pattern.
- **IRF Length:** Length of the IRF pattern to be used for reconvolution fitting.
- **IRF Shift:** Shift of the IRF pattern with respect to the fluorescence decay.
- **Perp IRF Shift:** Alignment of the IRF in the perpendicular channel with respect to the parallel channel IRF.
- **Scat Shift:** Shift of the scatter pattern with respect to the fluorescence decay.
- **Perp IRF Shift:** Alignment of the scatter pattern in the perpendicular channel with respect to the parallel channel scatter pattern.
- **Ignore Length:** Can be used to ignore a part of the data range for the calculation of the $\chi^2_{\text{red.}}$, while still using the omitted data range for reconvolution.

Generally, the IRF pattern has to be shifted to accurately fit the data using the reconvolution approach. This is caused by *countrate dependent timing shifts* of the APDs, since IRF and TCSPC measurement are usually performed at different count rates. It can be difficult to correctly assign the shift of the IRF pattern with respect to the fluorescence decay. The user can instead supply a range of IRF shifts and let **TauFit** find the optimal value (see the description of the *fit table*).

4.2.3 Correction factors

For accurate fluorescence lifetime and anisotropy analysis, a few correction factors have to be known. Most important for lifetime analysis is the so-called *G*-factor, accounting for differences in the detection efficiency η between the parallel and the perpendicular detection channel.

$$G = \frac{\eta_\perp}{\eta_\|}$$

G can be determined from the relative intensity levels from a measurement of an uncorrelated light source, i.e. a fluorescent dye with a short rotational correlation time. One can either directly take the ratio of the fluorescence signal at large lag time, or calculate *G* from the residual anisotropy r_∞ of the dye measurement (see *Time-resolved anisotropy fitting*). *G* is the given by:

$$G = \frac{1 - r_\infty}{2r_\infty + 1}$$

The button **Determine G factor** can be used to automatically determine *G* using the described method. The data range should be selected as described for tail fitting or time-resolved anisotropy analysis using the **Ignore Length** slider.

Additionally, anisotropy measurements with high numerical aperture objectives are subject to depolarization due to polarization mixing [Koshioka1995]. This causes a decrease of the measured anisotropy and is accounted for by the correction factors l_1 and l_2.

4.3 Basic data analysis

4.3.1 Tail fitting

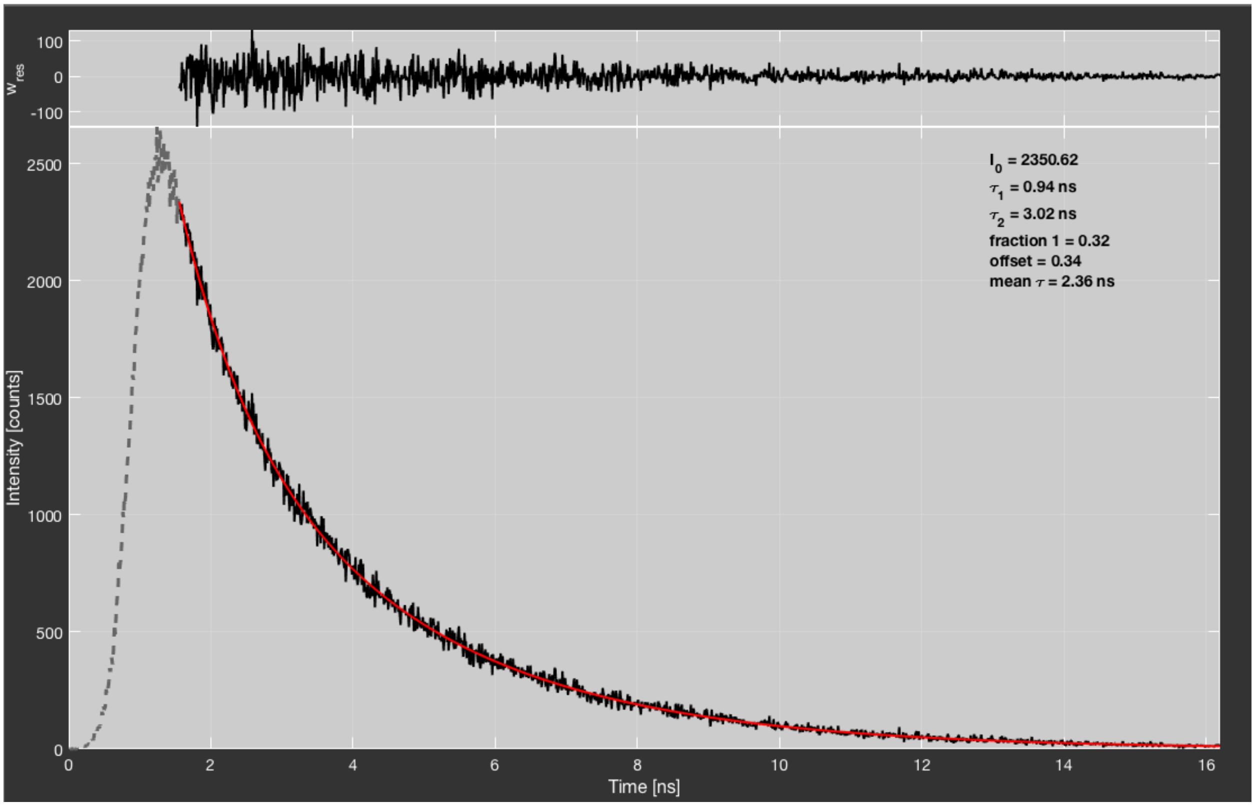


Fig. 4: Tail fitting using a biexponential model function.

For a quick determination of the fluorescence lifetime, a single or biexponential or triexponential tail fit of the fluorescence decay can be performed. The parallel and perpendicular channels should be aligned properly. Additionally, the initial rise of the fluorescence decay should be excluded either by using the **Para Start** or the **Ignore Length** slider. The fit result is displayed in the top right corner of the plot.

The fit model is given by:

$$F(\tau) = I_0 \left(\sum_{i=1}^{N} \alpha_i e^{-t/\tau_i} \right) + c$$

For fitting, the parallel and perpendicular fluorescence decays are combined such that the anisotropy information is canceled out.

$$D = (1 - 3l_2)\, G I_{||}(t) + (2 - 3l_1)\, I_{\perp}(t)$$

Tail fitting uses weighted residuals if specified in the options (**Use weighted residuals**).

Intensity vs. species fraction in TCSPC

When using multi-exponential fit models to describe TCSPC data, it is important to note that the obtained fractions α_i give the amplitudes at time *t = 0* and are proportional to the number of molecules of the respective lifetime. However, the intensity contribution of a species to the fluorescence decay is proportional to the area of the curve, given by $A_i = \tau_i \alpha_i$. The average lifetime of a multi-exponential is calculated based on the intensity fractions f_i given by:

$$f_i = \frac{\alpha_i \tau_i}{\sum_j \alpha_j \tau_j}$$

The average lifetime is then given by:

$$\langle \tau \rangle = \sum_i f_i \tau_i$$

This value is reported in the **Status Message** box in the bottom right, along with the respective intensity fractions.

On the contrary, the value of:

$$\tau_{\text{amp.}} = \sum_i \alpha_i \tau_i$$

is proportional to the area under the curve and thus the steady-state intensity. See [Lakowicz2007] for more details.

4.3.2 Time-resolved anisotropy fitting

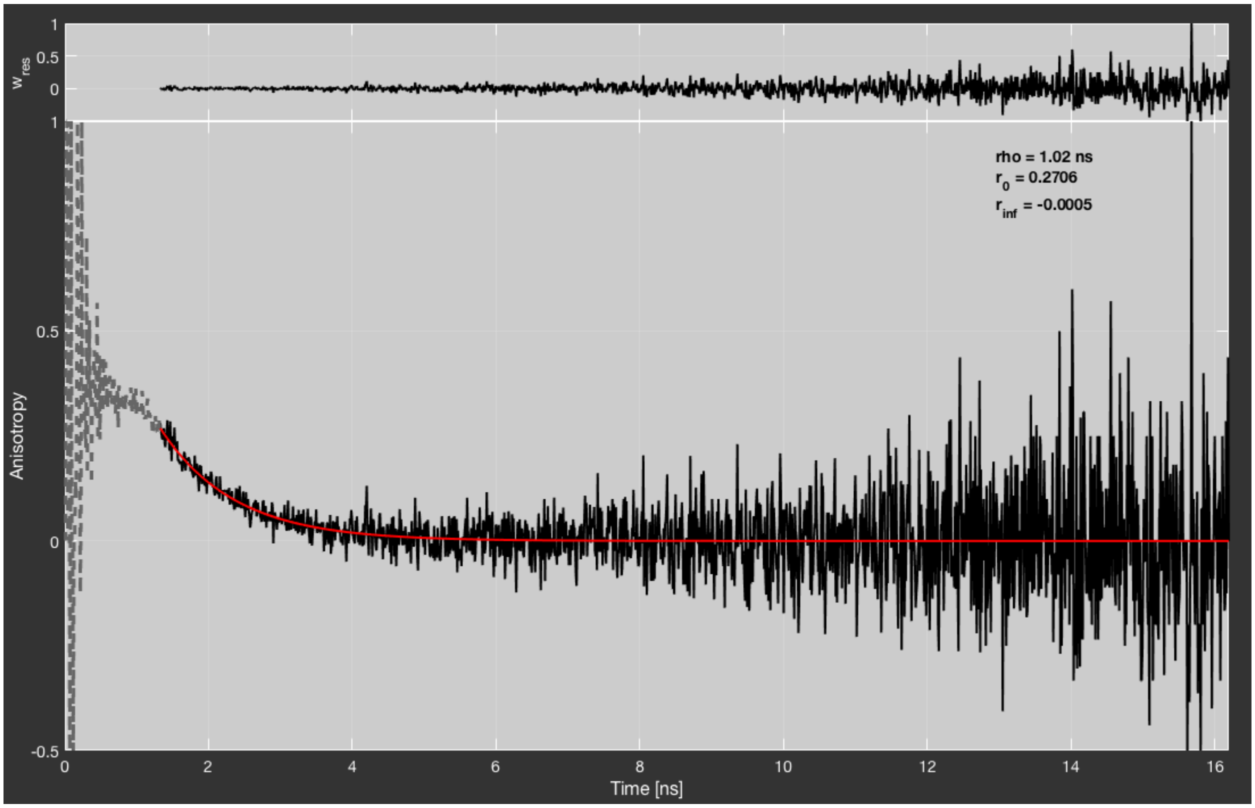


Fig. 5: Time-resolved anisotropy fitting.

Similar to tail fitting of fluorescence decays, the time-resolved anisotropy can be analyzed using basic fitting. The anisotropy decay is calculated using:

$$r(t) = \frac{GI_{\parallel}(t) - I_{\perp}(t)}{(1 - 3l_2)\, GI_{\parallel}(t) + (2 - 3l_1)\, I_{\perp}(t)}$$

The standard model function is based on the wobbling-in-cone approximation, which assumes that the fluorophore's rotation is restricted by the attachment to the biomolecule. It is given by:

$$r(t) = (r_0 - r_\infty)\, e^{-t/\rho} + r_\infty$$

where r_0 is the fundamental anisotropy of the fluorophore, ρ is the rotational correlation time describing the local motion, and r_∞ is the residual anisotropy due to the slow rotational movement of the biomolecule on a timescale larger than the fluorescence lifetime. If the rotation of the biomolecule ρ_p is faster, it can be described by an additional exponential term:

$$r(t) = \left[(r_0 - r_p)\, e^{-t/\rho_f} + r_p\right] e^{-t/\rho_p}$$

The biexponential fit model is available by *right clicking* the **Fit anisotropy** button.

Additionally, a model called "dip-and-rise" can be fitted. This model assumes two independent species with different fluorescence lifetimes and rotational behavior, where each component shows a mono-exponential anisotropy decay and single-component fluorescence lifetime. A "dip-and-rise" behavior is often observed for cyanine dyes, where a change in the rotational freedom is connected to the fluorescence lifetime, an effect known under the name *PIFE* for *protein induced fluorescence enhancement*. Generally, steric hindrance limits the *cis-trans* isomerization rates, resulting in a longer fluorescence lifetime, whereas the sterically free dyes show a short fluorescence lifetime. A superposition of both species will result in the "dip-and-rise" anisotropy decay, since the average anisotropy depends on the relative contributions of the two species. The fast-rotating low-anisotropy species contributes at small lag times, causing the anisotropy to decline fast initially. At longer lag times, mainly the slow-rotating high-anisotropy species contributes to the fluorescence signal, causing the anisotropy to rise again.

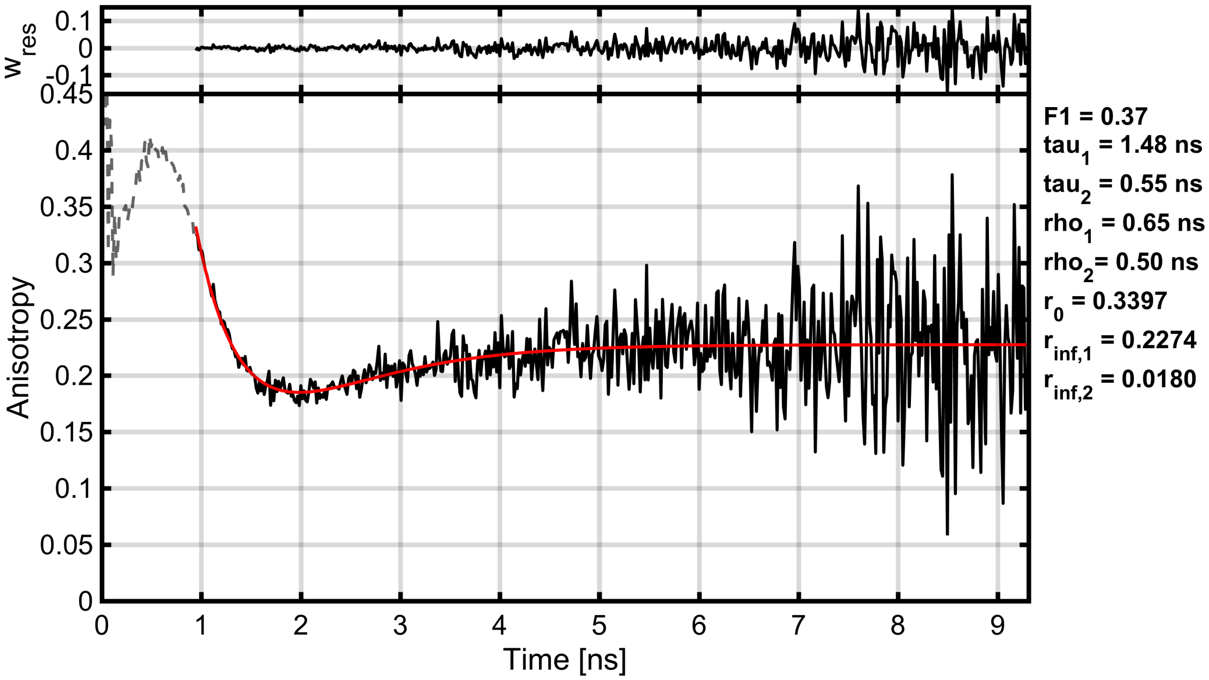


Fig. 6: "Dip-and-rise" behaviour of Cy3 labeled sample.

The model function is given by:

$$r(t) = A_1(t)\left[(r_0 - r_{\infty,1})\, e^{-t/\rho_1} + r_{\infty,1}\right] + (1 - A_1(t))\left[(r_0 - r_{\infty,2})\, e^{-t/\rho_2} + r_{\infty,2}\right]$$

Here, the amplitude of species 1 $A_1(t)$ is given by:

$$A_1(t) = \frac{I_{0,1}e^{-t/\tau_1}}{I_{0,1}e^{-t/\tau_1} + I_{0,2}e^{-t/\tau_2}} = \left(1 + \frac{I_{0,2}}{I_{0,1}}e^{-t\left(\frac{1}{\tau_2}-\frac{1}{\tau_1}\right)}\right)^{-1}$$

The model assumes identical fundamental anisotropy r_0 for both species. The returned parameter F_1 is the fraction of species 1 given by:

$$F_1 = \frac{I_{0,1}}{I_{0,1} + I_{0,2}}$$

4.4 Reconvolution fitting

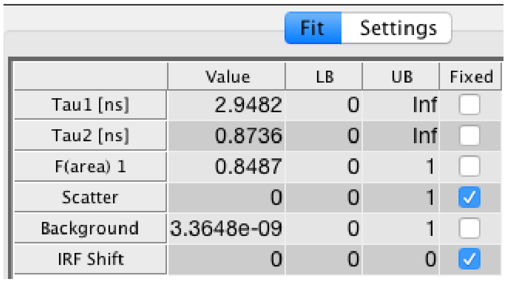
Reconvolution fitting uses the experimentally obtained instrument response function (*IRF*) to convolute the model function:

$$D(t) = IRF(t) * M(t)$$

Here, $*$ denotes the convolution operation. If polarized detection is used, the anisotropy-free fluorescence decay is constructed as described *above*. The IRF of the combined channel is then approximated in the same manner.

The **Convolution Type** can be changed in the **Settings** tab. *Linear* convolution performs convolution without periodicity. If the selected PIE channels spans the total available microtime range, i.e. the decay pattern is periodic, *circular* convolution should be used.

4.4.1 Fit table



Fit | Settings

	Value	LB	UB	Fixed
Tau1 [ns]	2.9482	0	Inf	□
Tau2 [ns]	0.8736	0	Inf	□
F(area) 1	0.8487	0	1	□
Scatter	0	0	1	☑
Background	3.3648e-09	0	1	□
IRF Shift	0	0	0	☑

Fig. 7: The fit parameter table.

The fit table shows the parameters of the currently selected **Fit Method**. Initial parameter values and lower and upper boundaries (**LB**, **UB**) can be specified here. Additionally, parameters can be fixed.

The **IRF Shift** parameter is a special case. If it is unfixed, the fit will iterate over a range of IRF shift values given by the specified center value and lower/upper bound, i.e. from `Shift-LB:1:Shift+UB`, and perform the fit routines at those shift values. The goodness of fit is compared between all fits, and the best fit attempt is returned.

4.4.2 Model parameters

There are several common parameters that are part of all model functions:

Background: Inclusion of a constant background contribution into the fit, i.e. a constant offset *c* with intensity fraction *b*:

$$D(t) = (1-b)IRF(t) * M(t) + bc$$

Scatter: Inclusion of a scatter contribution *S(t)* as defined from a buffer measurement with intensity contribution *s*:

$$D(t) = (1-s)IRF(t) * M(t) + sS(t)$$

Generally, one should consider using either a constant background if only uncorrelated background noise is present, or only a scatter microtime pattern from a buffer measurement if there is considerable background contribution from the excitation laser. Alternatively, one can check the **Scatter offset = 0** option in the **Settings** tab, which will remove the constant background contribution to the scatter pattern by substracting a baseline first, in which case both options can be used together.

To describe the fluorescence decay, the following model functions $M(t)$ are available:

Exponential decays

Identical to the tail fitting models (see *Tail fitting*), with up to three exponentials available.

Distance distribution

This option fits a Gaussian distance distribution model, given by:

$$M(t) = \int \frac{1}{\sqrt{2\pi}\sigma_R} \exp\left[-\frac{(R-R_{DA})^2}{2\sigma_R^2}\right] \times \exp\left[-\frac{t}{\tau_{D,0}}\left(1+\left(\frac{R_0}{R}\right)^6\right)\right] dR$$

A donor only contribution can be included:

$$M'(t) = (1-d)M(t) + d\exp\left(-t/\tau_D\right)$$

This model is used to determine the width of donor-acceptor distance distributions in FRET experiments which arises due to the attachment of fluorophores via flexible linkers. The determined distribution width can be used to determine the static FRET line in presence of linker fluctuations in BurstBrowser (see *static FRET line*).

Anisotropy models

These model functions fit the parallel and perpendicular intensity decays simultaneously with globally linked parameters. The model function incorporates both the intensity decay *I(t)* and the anisotropy decay *r(t)*:

$$M_{||}(t) = \left[1+(2-3l_1)r(t)\right]I(t)$$
$$M_{\perp}(t) = \left[1-(1-3l_2)r(t)\right]I(t)$$

Background and scatter contributions are separately applied to the model for the parallel and perpendicular decay. The model functions are then convoluted with the respective channel's IRF.

For the intensity decay *I(t)* mono- and biexponential models are available. For the anisotropy, the two models described in *Time-resolved anisotropy fitting* are available. Here, it is assumed that all molecules show identical behavior.

Additionally, the *dip-and-rise model*, which describes two species with different lifetimes and different rotational properties, is available as *Fit anisotropy (2 exp lifetime with independent anisotropy)*.

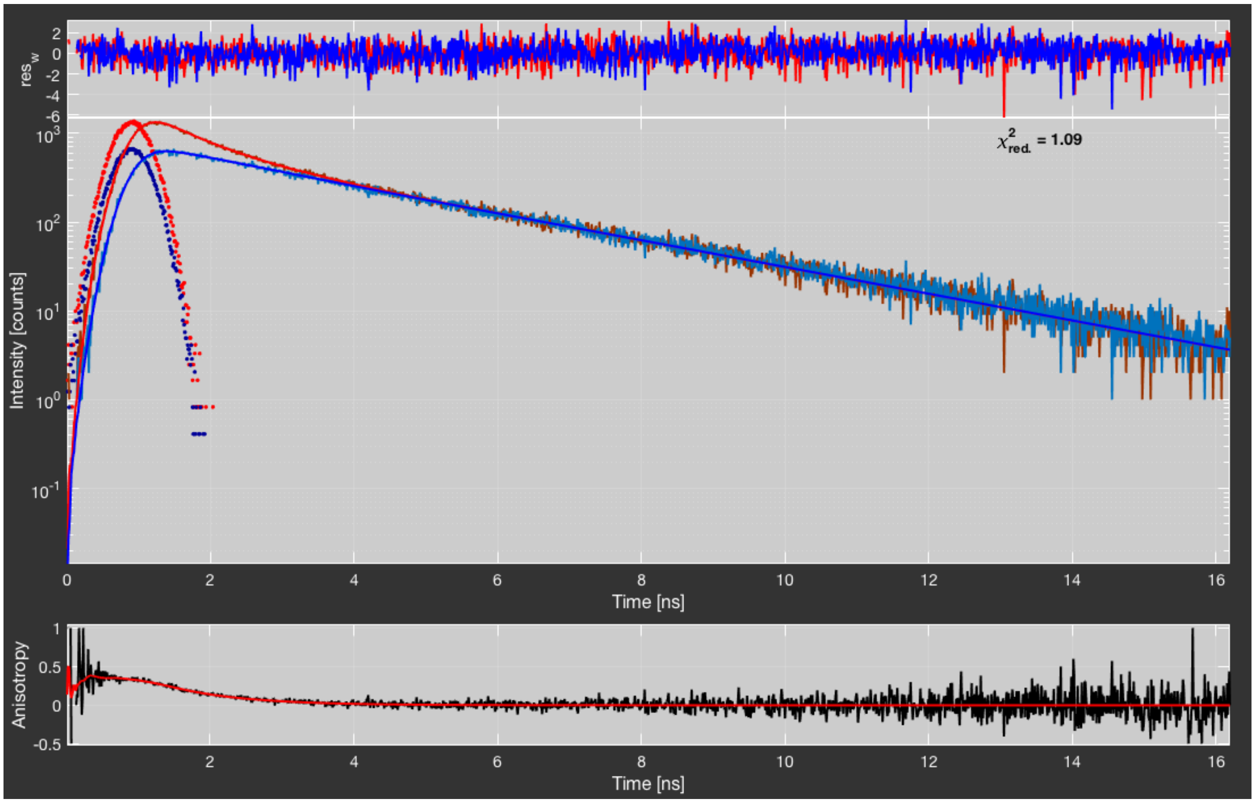


Fig. 8: Fitting using an anisotropy model.

When using an anisotropy model, additionally the experimental and fitted anisotropy will be displayed below the main plot.

4.4.3 Extrapolation of the instrument response function

The experimentally determined instrument response function (IRF) might suffer noise due to low statistics, or be contaminated by a fluorescent impurity that comprises the ideal IRF pattern. In these cases, it is helpful to extrapolate the IRF using an appropriate model function. That model function is usually a gamma distribution, which is given by:

$$\text{Gamma}(x|\alpha,\beta,s,b) = \frac{\beta^\alpha}{\Gamma(\alpha)}(x-s)^{\alpha-1}e^{-\beta(x-s)} + b$$

Here, the gamma distribution is given in the shape-rate parametrization with shape α and rate β, $\Gamma(\alpha)$ is the complete gamma function, and s and b are parameters accounting for the shift of the IRF pattern and constant background.

The IRF pattern can be fit to the outlined model before the reconvolution fitting procedure is performed. Enable this option in the *Settings* tab by checking the *Clean up IRF by fitting to Gamma distribution* checkbox. The plot below the checkbox is used to show the result of the fitting procedure, whereby the measured IRF pattern is shown as blue dots, and the fitted model is given as a red line. The IRF pattern will automatically be fit to the model when the fit is started using the *Perform reconvolution fit* button.

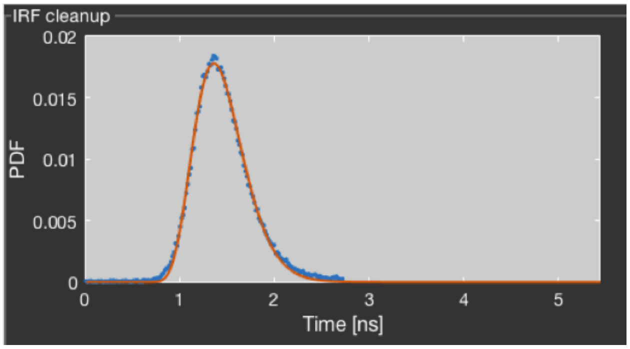


Fig. 9: Extrapolation of the IRF using the Gamma distribution.

4.5 Simulation of κ^2 distributions

When loading species-wise fluorescence decay data from BurstBrowser, additional functionality is available to determine the distribution of κ^2 values from the available fundamental anisotropies of the donor and acceptor, as well as residual anisotropies of donor, acceptor, and FRET sensitized acceptor emission [Ivanov2009]. The program reports the average value with standard deviation. Additionally, the accuracy of determined distances (i.e. the deviation from the true distance value if $\kappa^2 = 2/3$ was assumed) and the precision (i.e. the statistical uncertainty of the determined distance) is reported.

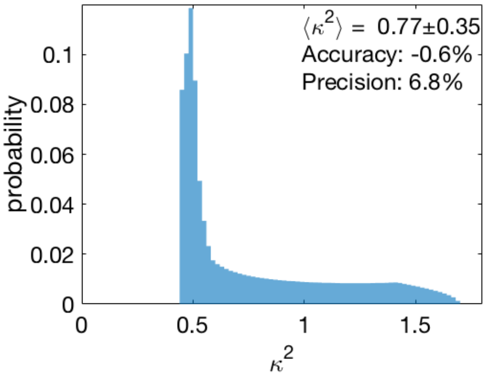


Fig. 10: Simulated κ^2 distribution using $r_0(D) = 0.4$, $r_0(A) = 0.4$, $r_\infty(D) = 0.1$, $r_\infty(A) = 0.1$ and $r_\infty(DA) = 0.05$.

4.6 Additional settings and exporting

Use weighted residuals: Tail fitting and reconvolution fitting support the use of weighted residuals. If **Use weighted residuals** is unchecked, the fit quality will be judged based on the mean squared deviation only.

$$\text{Mean squared deviation} = \frac{1}{N-P-1}\sum_i (x_i - f_i)^2$$

Here, x_i are the data points, f_i is the model function, N is the number of data points and P is the number of fit parameters (i.e. $N - P - 1$ is the number of degrees of freedom). If **Use weighted residuals** is checked, the error for each data point σ_i will be estimate assuming Poissonian counting statistics, i.e. $\sigma_i = \sqrt{x_i}$. The reduced chi-squared value $\chi^2_{\text{red.}}$ is the maximum likelihood estimate for Gaussian errors and is defined by:

$$\chi^2_{\text{red.}} = \frac{1}{N-P-1}\sum_i \frac{(x_i - f_i)^2}{\sigma_i^2}$$

$\chi^2_{\text{red.}} \approx 1$ indicates a good fit, while $\chi^2_{\text{red.}} >> 1$ means that the model is not sufficient to describe the data. $\chi^2_{\text{red.}} < 1$ is an indication for overfitting, meaning that the complexity of the model should be reduced.

For anisotropy fitting, the usage of weighted residual is not available, since no straight-forward estimate is available for the estimation of the error of the calculated anisotropy.

Automatic fit update: Automatic fitting can be enabled in the **Setting** tab by checking the **Automatic fit** checkbox. This will cause the fit to automatically update whenever sliders are changed.

y-axis scale: The y-scaling of the main plot can be changed between *linear* and *logarithmic* by *right clicking* the main plot and selecting **Y Logscale**.

x-axis scale: The x-scaling of the main plot can be changed between *linear* and *logarithmic* by *right clicking* the main plot and selecting **X Logscale**.

Exporting plots: Plots can be exported by clicking **Export Plot** in the right-click menu, which will open the current plot in a new MATLAB figure.

Exporting data:

- The fluorescence decay or anisotropy data can be exported as a *.txt* file by clicking **Export -> Save Data to .txt**. This will create a file with ending **_tau.txt* or **_aniso.txt*, depending on what kind of data is plotted. To compare exported data, click **Export… -> Compare Data** and select the exported *.txt* files. This will automatically create a comparison plot. When comparing fluorescence decays, the data will be aligned with respect to the maximum intensity point.
- Alternatively, the currently plotted data can be copied to the clipboard, so it can easily be pasted into other software like Excel or Origin (**Export… -> Copy Data to Clipboard**).
- TauFit integrates with the filtered-FCS functionality of PAM to generate synthetic microtime patterns. This is supported only when using an anisotropy model. Perform a fit and go to **Export… -> Export fitted microtime pattern** to export the current fit as *.mi* file that can be used in filtered-FCS analysis.

CHAPTER 5

Microtime Image Analysis (MIA)

Contents

The MIA window can be opened from the PAM main window by clicking on **Advanced Analysis -> MIA** or from the command window by typing in `Mia`.

MIA is used for a variety of different image analysis methods, mainly via image correlation techniques.

Sample data for different types of analyses can be found here: https://gitlab.com/PAM-PIE/PAM-sampledata

An introduction and description of the various techniques can be found in:

Image Correlation Spectroscopy (ICS): [Petersen1993]

Image Cross-Correlation Spectroscopy (ICCS): [Wiseman2000]

Raster Image Correlation Spectroscopy (RICS): [Digman2005] and [Digman2009a]

Spatio-Temporal Image Correlation Spectroscopy (STICS): [Toplak2012]

Image Mean-Square Displacement (iMSD): [DiRienzo2013]

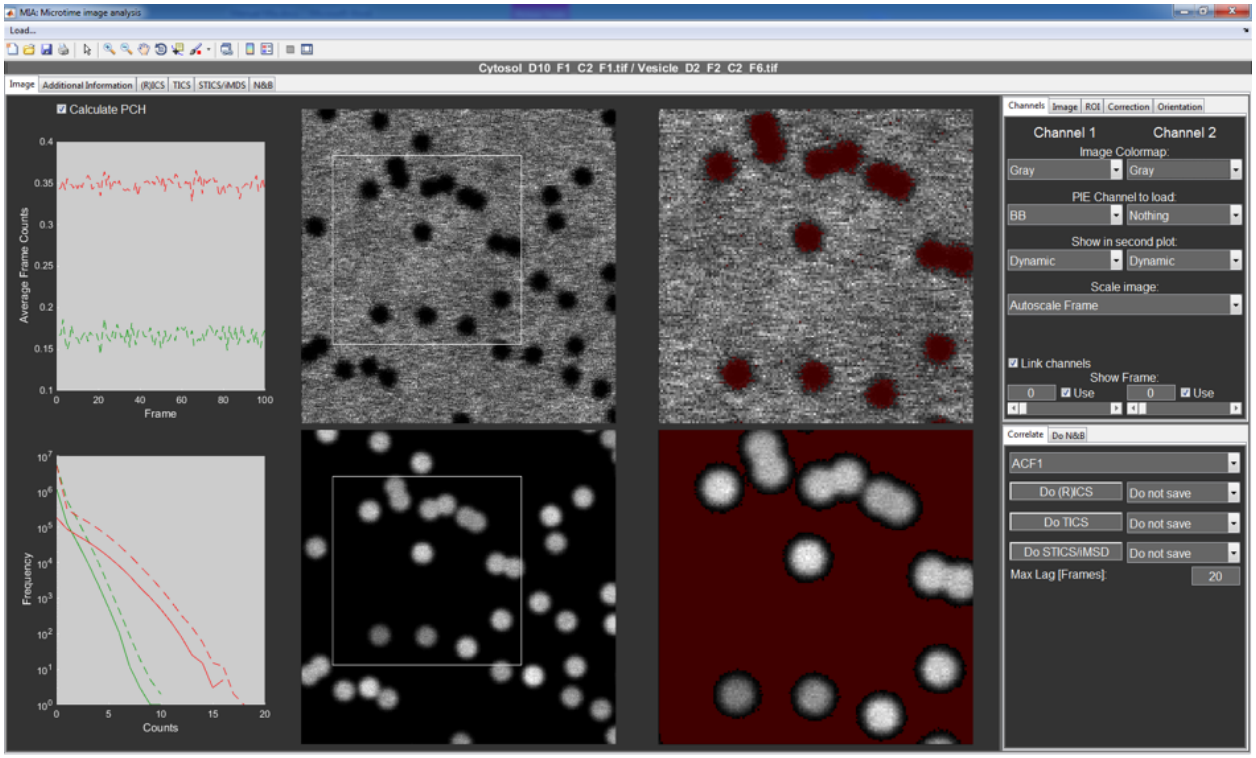
Number & Brigthness (N&B): [Digman2008] and [Digman2009b]

Combination of Pulsed Interleaved Excitation and Fluctuation Imaging (PIE-FI): [Hendrix2013] and [Hendrix2015]

Arbitrary Regions for Image Correlation Spectroscopy (ARICS): [Hendrix2016]

Spectrally resolved RICS: [Schrimpf2018]

Performance evaluation for RICS: [Longfils2019]



Cytosol D10 F1 C2 F1.tif / Vesicle D2 F2 C2 F6.tif

☑ Calculate PCH

Channel 1 / Channel 2
Image Colormap: Gray / Gray
PIE Channel to load: BB / Nothing
Show in second plot: Dynamic / Dynamic
Scale image: Autoscale Frame
☑ Link channels
Show Frame: 0 ☑ Use / 0 ☑ Use

ACF1
Do (R)ICS: Do not save
Do TICS: Do not save
Do STICS/iMSD: Do not save
Max Lag [Frames]: 20

Fig. 1: MIA main gui

5.1 Data loading and handling

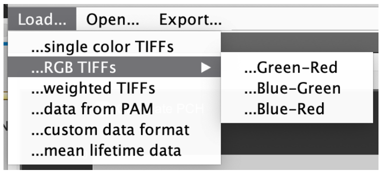


Fig. 2: The loading menu

MIA supports different kinds of input data.

Menu **Load…**

single color TIFFs: A popup window appears to select grayscale TIFF files to be loaded (multiple files will be concatenated). The file selection window will be opened a second time to select the files for the second channel. Press cancel if only one color is needed.

RGB TIFFs: A popup window appears to select RGB TIFF files to be loaded (multiple files will be concatenated). Depending on which menu item is chosen, the RG, RB or GR channels from the TIFF file are loaded.

data from PAM: Accessing the files currently loaded in PAM to generate an image series, using the PIE channels defined in the **Channels** tab.

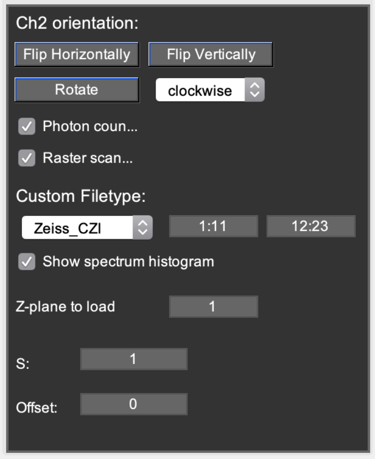
weighted TIFFs: For loading lifetime-resolved [Hendrix2013] or spectrally-resolved [Schrimpf2018] data. This works the same as standard TIFF data, but extracts re-scaling parameters from the data to account for the non-integer values saved with an integer file format. RLICS and RSICS TIFFs contain the unfiltered data in the first half and the filtered frames in the second half. This option loads only the filtered frames (the raw data can be loaded with the normal **single color TIFFs** option).

custom data format: This case depends on the particular format selected. To change the custom read-in to the desired format, go to the *Options tab*. and change the *Custom Filetype* dropdown menu.

If you have analog data, you can click the y axis label in the count rate plot to display the axis in counts instead of absolute photon counts (kHz). Alternatively, you can *convert analog to photon counting images* or select the analog checkbox on on the Options tab.

All data is stored in the global variable *MIAData* and can be easily accessed from the MATLAB command line.

5.1.1 Options tab



clockwise

☑ Photon coun...

☑ Raster scan...

Custom Filetype: Zeiss_CZI 1:11 12:23

☑ Show spectrum histogram

Z-plane to load 1

S: 1

Offset: 0

Fig. 3: The options tab

Miscellaneous options related to data loading and conversion prior to any analysis are grouped here.

Ch2 orientation: This option can be used to flip or rotate an image in case e.g. cameras were rotated with respect to each other.

Photon counting: Checking it will display all count rate values in kHz. Unchecking it will display counts in arbitrary units.

Raster scanning: Checking it will display the **pixel dwell time** and **line time** on the **Image** tab and converts pixel counts to kHz when calculating count rates. Unchecking it does not rescale pixel counts when calculating count rates.

Custom filetype: Here, the user can define specialized read-in routines. For Zeiss .czi files, use the Zeiss-CZI option. In the leftmost (rightmost) editbox, define the spectral channels summed into the top (resp. bottom) image channel of Mia. **1:11** means summing up spectral channels 1 to 11. The **Show spectrum histogram** option can be used to verify in which spectral windows the sample fluoresces.

Z-plane to load (only visible when the Zeiss-CZI option is selected): allows extracting specific planes from a multi-plane .czi file. You can also define a range of z-planes by writing e.g. 1:5 or 1,2,5. To allow correct blockwise RICS analysis of a z-stacks, Mia will automatically adjust the frame range (ROI tab) and Window/Offset on the Correlate tab.

S & Offset: Here, the parameters from [Dalal2008] can be entered to *convert analog to photon counting images*.

5.2 Main Image Tab

The central controls and settings of **MIA** are found in the **Image** tab. The two leftmost plots show the mean frame intensity and a photon counting histogram. The green lines correspond to the first and the red to the second channel. The dotted lines represent the data for the full stack, while the solid line represents the selected pixels in the region of interest (ROI) after correction. A mouse click on the y-label toggles different displaying options.

The four images in the center show the full frame (left) and the selected and corrected ROI (right) for the first (upper images) and second (lower images) channels. On the leftmost images, the ROI is drawn via a solid square or rectangle.

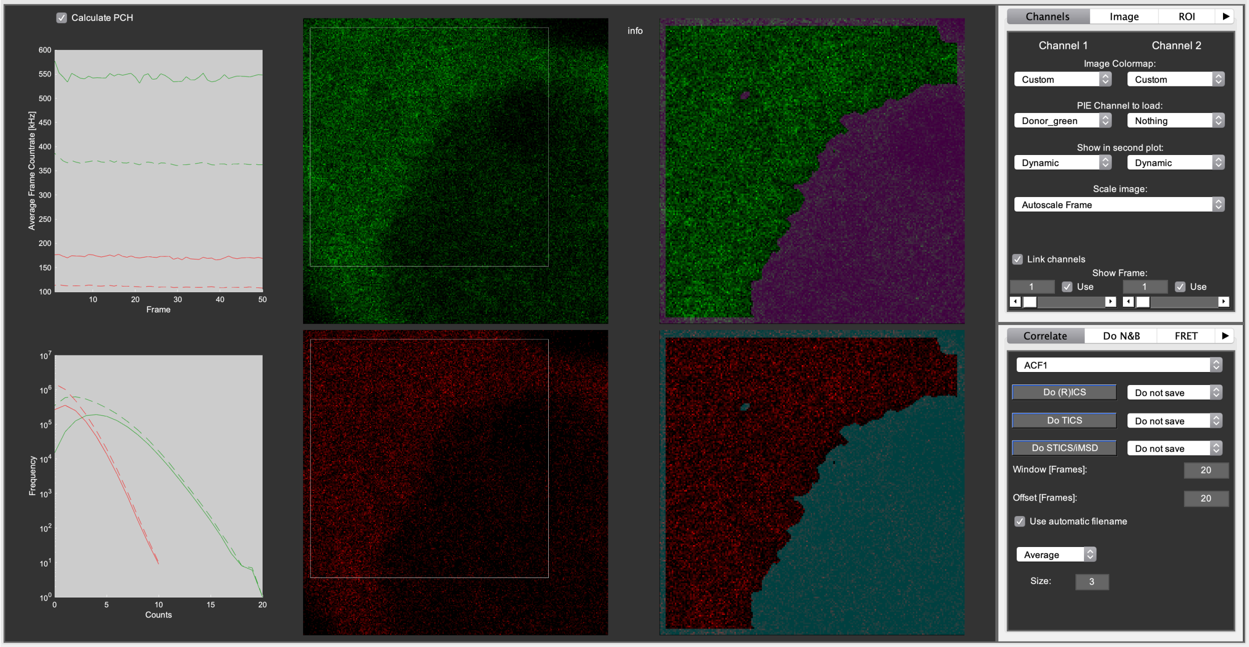


Image Colormap: Custom | Custom

PIE Channel to load: Donor_green | Nothing

Show in second plot: Dynamic | Dynamic

Scale image: Autoscale Frame

☑ Link channels

Show Frame: 1 ☑ Use | 1 ☑ Use

ACF1

Do (R)ICS: Do not save

Do TICS: Do not save

Do STICS/iMSD: Do not save

Window [Frames]: 20

Offset [Frames]: 20

☑ Use automatic filename

Average

Size: 3

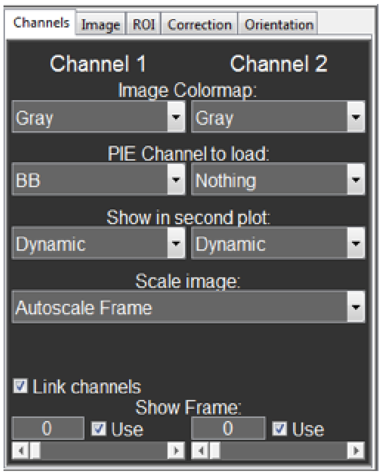
Fig. 4: The image tab

Hint: All images in MIA can be **exported** as 8bit RGBs with the same contrast as displayed.

- **Leftmost images:**
  - "left-click": center ROI
  - "ctrl"+"left-drag" or "right-drag": draw **normal ROI**
  - "shift"+"left-click" or "middle-click": export image.
  - if the 'Left Image Context Menu' checkbox on the ROI tab is checked, an **arbitrary ROI** can be drawn via "right-click" on the image
- **Rightmost images:**
  - "right-click": draw AROI popupmenu
  - "shift"+"left-click" or "middle-click": export image
  - "right-click" on the rightmost images gives a context menu for manually drawing an arbitrary ROI using "left-drag"
- **Dual color images or videos:**
  - Can be generated using the "Export" Menu

All settings for displaying, correcting and analyzing the images can be found in the various tabs at the right of the window, and will be individually discussed in the next sections.

5.2.1 Channels



Channel 1 Channel 2

Image Colormap:
Gray Gray

PIE Channel to load:
BB Nothing

Show in second plot:
Dynamic Dynamic

Scale image:
Autoscale Frame

☑ Link channels

Show Frame:
0 ☑ Use 0 ☑ Use

Fig. 5: The channels tab

This tab contains mainly setting for displaying the images and switching between frames. From top to bottom these are:

Image Colormap: This changes the colormap for the channels between different options to standard MATLAB colormaps (Gray, Jet, Hot and HSV). The HiLo colormap is the same as the Gray colormap with the exception that all pixels below and above the manual scaling are displayed in blue and red, respectively. The Custom colormap uses a single color (standard green and red) for displaying. The color can be changed with the middle mouse button.

PIE Channel to load: This function selects the PIE channel that will be used when data is loaded from the **PAM** main window. For the second channel *Nothing* can be selected for one channel analysis.

Show in second plot: This option switches between different options to display in the right images. *Original* and *Dynamic* show the ROI before and after the corrections are applied, respectively. *Static* show the corrections that were applies, i.e. the difference between *Original* and *Dynamic*.

Scale Image: This switches between different scaling options. *Autoscale Frame* scales the image to the min/max of the displayed image, while *Autoscale Total* scales to the min/max of the full stack. *Manual Scale* allows the users to set their own fixed scaling limits.

Link channels: This checkbox (un)links the two channels, so that the frames can be changed individually or together.

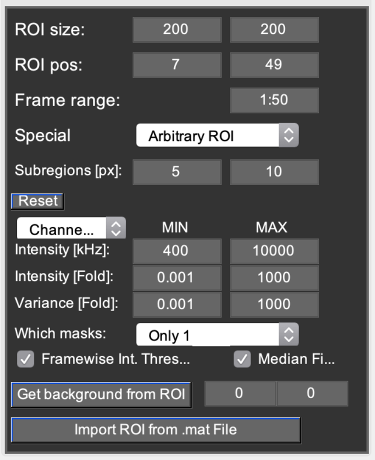
Show Frame: These edit boxes and sliders are used to switch between different frames of the stack. When frame "0" is selected, the program displays the mean of **Frame range** displayed on the **ROI** tab. With the **Use** checkbox the user can unselect individual frames for some further analysis.

5.2.2 Image

This tab contains the most important information about the image stacks, like frame, line and pixel time and the pixel size. These values are used for certain calculations and data processing. If the metadata contain information on any of these parameters, these values will be updated automatically.

Scale bar: Enter any value to display the scale bar on the images. If the entered value is too large, Mia will automatically make the scale bar fit within both the left and right images. Enter 0 or no value to hide the scale bar. Scale bars are included in the exported images.

5.2.3 ROI



ROI size: 200 200

ROI pos: 7 49

Frame range: 1:50

Special: Arbitrary ROI

Subregions [px]: 5 10

Channe...	MIN	MAX
Intensity [kHz]:	400	10000
Intensity [Fold]:	0.001	1000
Variance [Fold]:	0.001	1000

Which masks: Only 1

☑ Framewise Int. Thres... ☑ Median Fi...

Get background from ROI: 0 0

Fig. 6: The ROI tab

This tab contains options for selecting ROIs and discarding pixels vie arbitrary region ICS. From Top to bottom:

ROI size and position: These edit boxes define the size and position of the selected ROI for both channels. Alternatively, the user can click on the left images to position the ROI. Right-dragging the leftmost images also allows drawing the ROI box. Left clicking shifts the center of the current ROI without changing the size.

Frame range: This allows the user to define the frames used for most further data processing. This uses standard MATLAB notation for array slicing. When frame 0 is selected on the %%Channels%% tab, Mia displays the average image calculated from this frame range.

***Special* popupmenu:** This popupmenu switches between (1) using all frames within the frame range, (2) using the frames within the frame range that have a checked **Use** checkbox on the **Channels** tab, or (3) using arbitrary region ICS for further data processing. Arbitrary region ICS ([Hendrix2016]) allows discarding individual pixels from the correlations to remove the influence of unwanted objects and artifacts from the analysis. There are different ways of unselecting pixels, which can be combined.

Subregions [px]: Further options below employ the mean property of pixels in one or two small ROIs within the image. The size of those small ROIs is defined by these two edit boxes. The left size always has to be smaller than the right size.

***Reset* pushbutton:** This button resets all AROI selection values to their defaults.

***Channel* popupmenu:** Allows switching between the AROI values for channel 1 or channel 2, as these will be different.

Intensity [kHz]: Pixels are discarded if their mean intensity throughout the whole stack falls outside the range defined by MIN and MAX. A value equal to zero for MIN and MAX turns it off. This is especially useful for discarding e.g. the cell exterior.

Intensity [fold] or Variance [fold]: The intensity or intensity variance inside the small **Subregions** ROI is compared with the large **Subregions** ROI. If the ratio between those values is below or above the range defined by the values in the edit boxes, the pixel is discarded. For pixels at the edges the ROI cannot be created. These pixels are therefore automatically discarded. Note: These options are performed on a frame to frame basis, so pixels are not automatically removed from the whole stack. Pixels that use discarded pixels for a moving average subtraction/addition are also discarded.

***Which masks* popupmenu:** Independent: Masks are generated per channel. Note: resulting analysis may be performed for different pixels in channel 1 and 2.

Only 1 (2): The mask for channel 1 (resp. 2) is automatically applied to channel 2 (resp. 1), the other channel's mask is ignored. 1&2: Only pixels included into the mask of both channels are kept. All other pixels are excluded from the mask.

Framewise Int. Threshold: Pixels are discarded on a framewise basis if their mean intensity in the small **Subregion** falls outisde the range defined by MIN and MAX. Note: The intensity in the small subregion is rescaled (/(mean int.frame)*(mean int.frame1) to account for intensity fluctuations such as drift or photobleaching.

Median Filter: Smoothens the arbitrary ROI by applying a 3x3 median filter.

For all arbitrary region ROIs, unselected pixels are highlighted in the right images, based on the selected colormap.

An arbitrary ROI can also be manually drawn onto any image. Once **Arbitrary ROI** has been selected in the **Special** popupmenu, Right-clicking the corrected image will display a menu. When **Select manual ROI** is chose, holding down and dragging with the left mouse button will draw an AROI on the image. The selection applies throughout the stack. Drawing an AROI on the left image is also possible by checking the 'Left Image Context Menu' checkbox.

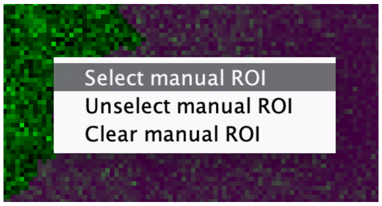


Fig. 7: The AROI drawing menu

***Get background from ROI* pushbutton:** For some applications (e.g. RICS or N&B) it might be useful to subtract the background from the corrected image prior to further analysis. Pushing this button displays the average count rate (in photons per pixel dwell time) in the currently selected ROI. Alternatively, if you know the background count rate from a control experiment, you can also write the value directly in the edit box. The left edit box is channel 1, the right one is channel 2. The values displayed in the edit boxes are subtracted from the images after any correction (see Correction tab) has been carried out.

***Import ROI from .mat file* pushbutton:** Allows importing a mask from a pre-defined .mat file. This file can be generated manually, or in the N&B tab. It has to have the following content: % ROI = [sizeX sizeY posX PosY] % Mask = ROI-sized single frame mask

5.2.4 Correction

This tab has options to apply corrections to the data, e.g. to get rid of an immobile fraction for (R)ICS. It includes various options for subtracting from and adding to the image, including a moving average (both pixels and frames), and pixel, frame and total mean. For the calculation of these values the discarded pixels from the arbitrary region selection are not used.

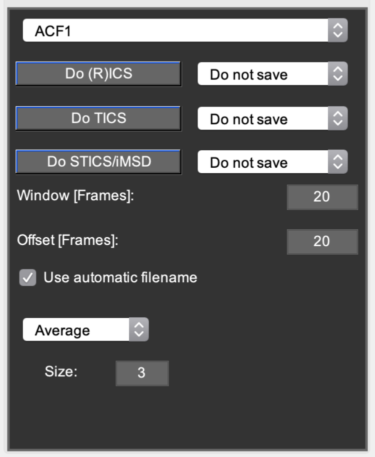
For RICS analysis, typically a pixel-based 3-frame (from frame-1 to frame+1) moving average subtraction (Pixels: 1 and Frames: 3) and a total mean addition are typically used [Hendrix2015]. Correlation amplitude bias due to this correction is undone [Hendrix2016].

For N&B analysis, typically a box-based moving average subtraction (e.g. Pixels: 3 and Frames: 3) and a pixel mean addition are typically used [Hendrix2015].

For TICS analysis, typically a frame mean subtraction and a total mean addition are typically used to correct for slow intensity drift [Hendrix2015].

Drift correction: Correction for linear lateral drift of the sample. When checked, Mia cross-correlates every frame with the first one and using the (x,y) position of each of the crosscorrelation functions, it laterally translates each frame relative to the first one. A graph of the x,y offset of the crosscorrelations is also plotted.

5.2.5 Correlate tab



Window [Frames]: 20

Offset [Frames]: 20

☑ Use automatic filename

Size: 3

Fig. 8: The Correlate tab

At the bottom right of the main window is a tab that contains all settings and buttons for applying the various ICS methods like (R)ICS, TICS, STICS and iMSD.

In the first popupmenu, the user can choose to calculate only a single autocorrelation function for the selected channel, or both auto- and crosscorrelation functions. Additionally, the user can choose for each method whether and how to save the calculated data.

Two extra edit boxes are present: *Window* and *Offset*. (R)ICS using the **'save blockwise analysis as .miacor'** option.

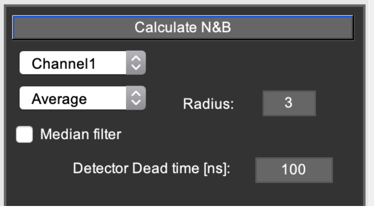
- *Window* is the window size (in frames) in which correlation data need to be averaged
- *Offset* is the offset (in frames) of the next window with respect to the previous.
- **For analyzing adjacent windows, *Offset* = *Window*. For example:** W=50, O=50 analyses the data from frame 1-50, 51-100,... W=50, O=10 analyses the data from frame 1-50, 11-60,...
- **Blockwise analysis can also applied to individual z-slices from .czi files containing z-stacks. For example:** Suppose there's 5 z-slices and 50 frames per slice, and a moving average correction of 3 frames is used. On the ROI tab in frame range, take 2:250 instead of 1:250 On the Correlate tab: W=48, O=50 analyses the data from frame 2-49, 52-99,..., this way effectively omitting the wrongly corrected frames 50 and 51 in between different slices. Suppose there's 5 z-slices and 50 frames per slice, and a moving average correction of 4 frames is used. On the ROI tab in frame range, take 2:250 instead of 1:250 On the Correlate tab: W=47, O=50 analyses the data from frame 2-48, 52-99,..., this way effectively omitting the wrongly corrected frames 49-51 in between different z-slices.

STICS/iMSD

- *Window* is the maximum number of lags (in frames) that have to be calculated.

The popupmenu: 'None', 'Average', 'Disk', 'Gaussian'. Spatially averages TICS correlations to improve statistics. The size (in pixels) can be entered in the respective checkbox

5.2.6 N&B tab



Channel1

Average

Radius: 3

□ Median filter

Detector Dead time [ns]: 100

Fig. 9: The N&B tab

The popupmenu: 'None', 'Average', 'Disk', 'Gaussian'. Applies an averaging filter to the intensity and variance images used for N&B analysis. The size (in pixels) can be entered in the respective checkbox.

Median filter. Applies a median filter to the resulting N and epsilon images on the **N&B** tab

Detector dead time [Hendrix2013] Enter a value here if a detection system (e.g. TCSPC or APD detectors) with non-negligible dead time is used for aqcuiring the data.

5.2.7 FRET tab

Used to calculated acceptor/donor FRET. Temporary version.

5.2.8 Co-localization tab

Used to calculate the Pearson's colocalization. Temporary version.

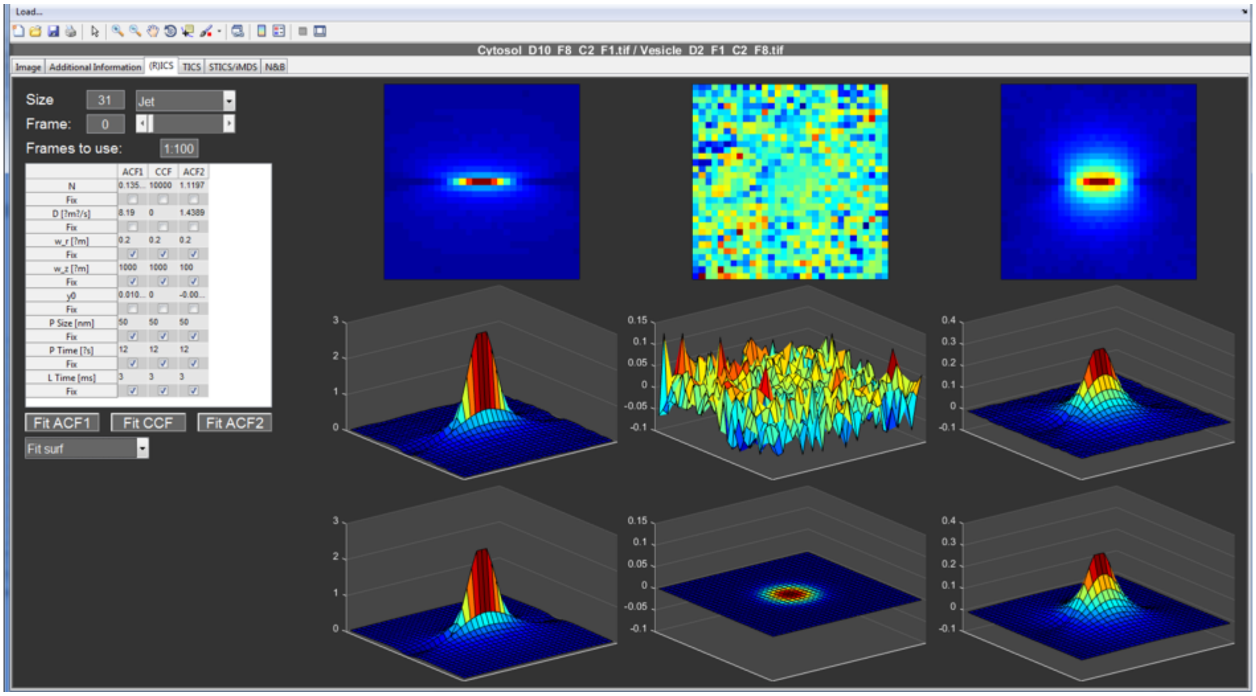
5.3 Additional information tab

This tab contains a couple of plots that show general information about the loaded files in greater detail. The left hand side shows the data as a frame average, while the right side shows the time average.

5.4 (R)ICS tab

The third window is reserved for displaying the image correlation data and a quick first evaluation of the data quality. Proper analysis is done using the **MIAFit** sub-program by selecting to save the data as a MATLAB file with a *.miacor* ending. It is also possible to save the data as TIFFs to load and plot them with standard image analysis programs like ImageJ.

The autocorrelations are plotted in the right and left plots, while the crosscorrelation is shown in the center plots. Hereby, the first row displays the correlations as and image plot. In the second row, it is plotted as a surface plot. The last row displays data related to the fit, either the fit or the residuals as a surface plot. The third option plots the data as a surface and the residuals in blue and red. The last option shows the on-axis data and fit.



Cytosol_D10_F8_C2_F1.tif / Vesicle_D2_F1_C2_F8.tif

Size: 31

Jet

Frame: 0

Frames to use: 1:100

	ACF1	CCF	ACF2
N	0.135...	10000	1.1197
Fix	□	□	□
D [?m?/s]	8.19	0	1.4389
Fix	□	□	□
w_r [?m]	0.2	0.2	0.2
Fix	☑	☑	☑
w_z [?m]	1000	1000	100
Fix	☑	☑	☑
y0	0.010...	0	-0.00...
Fix	□	□	□
P Size [nm]	50	50	50
Fix	☑	☑	☑
P Time [?s]	12	12	12
Fix	☑	☑	☑
L Time [ms]	3	3	3
Fix	☑	☑	☑

Fit surf

Fig. 10: The (R)ICS tab

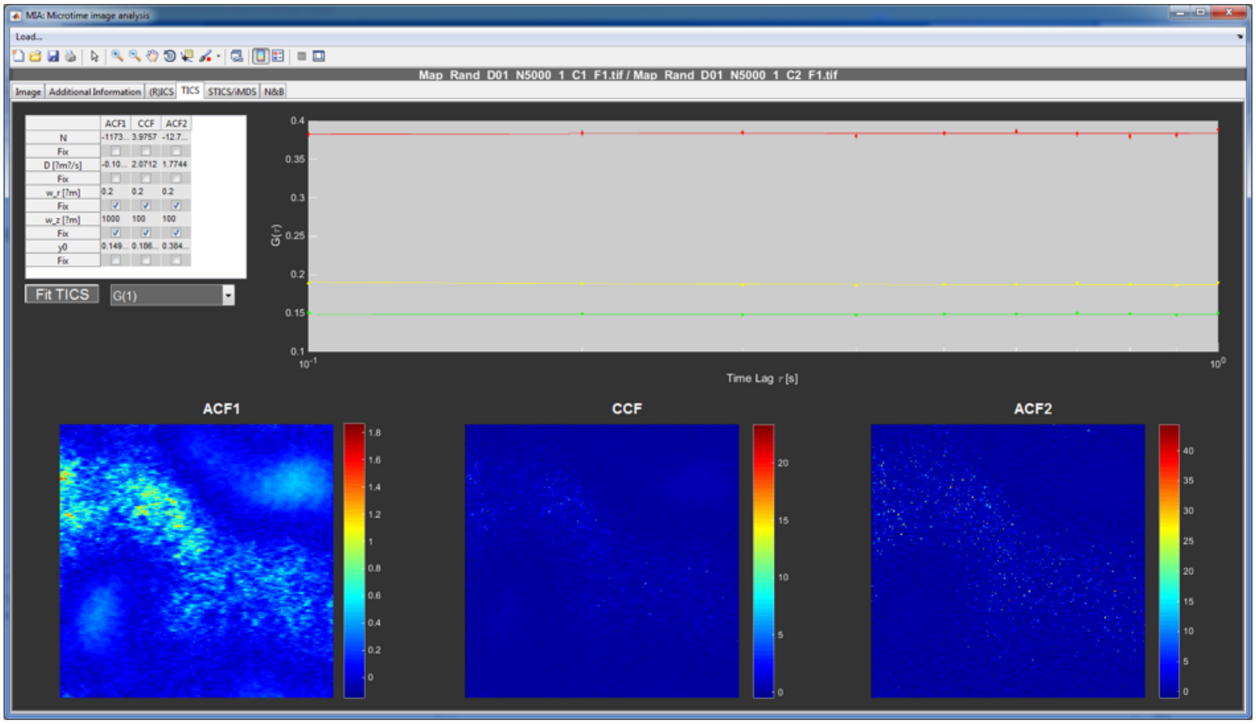
The setting in the top left define the size of the correlation to be used, the colormap for plotting and which frame to display. When frame "0" is selected, the program displays the average of the frames defined in the **Frames to use** edit box. Use standard MATLAB notation.

5.5 TICS tab

The TICS window is reserved for temporal correlation data and for a quick first evaluation of the data quality. Proper analysis is done using the **FCSFit** sub-program by selecting to save the data as a matlab file with a .mcor ending. To limit the contribution of undersampling, the correlation is only calculated up to one tenth of the maximum movie length.

The plot at the top of the window shows the pixel averaged auto- (green and red) and crosscorrelation (yellow). The table on the left can be used to perform a very simple 3D diffusion FCS fit.

The three images at the bottom show different calculated correlation parameters for the individual pixels for the two auto- (left and right) and the crosscorrelation (center). The user can select regions of the image by right clicking on them and export the averaged correlation for only the selected region.



	ACF1	CCF	ACF2
N	-1173...	3.9757	-12.7...
Fix	□	□	□
D [?m?/s]	-0.10...	2.0712	1.7744
Fix	□	□	□
w_r [?m]	0.2	0.2	0.2
Fix	☑	☑	☑
w_z [?m]	1000	100	100
Fix	☑	☑	☑
y0	0.149...	0.186...	0.384...
Fix	□	□	□

Fig. 11: The TICS tab

In the latest version of the TICS tab, different thresholding options are included:

G(1)-G(end): threshold on the basis of the amplitude of the decaying part of the TICS correlation

Brightness: threshold on the basis of Counts*(G1-Gend)

Counts: threshold on the basis of the average pixel intensity. Similar to arbitrary ROI thresholding.

Half-Life: threshold on the basis of the half-amplitude correlation time. This can be used to remove quasi-immobile pixels and other diffusive outliers from the average correlation function

**Samples: threshold on the basis of the number of times each lag of the correlation function for a particular pixel is sampled. This is important when an arbitrary ROI is used for generating the TICS data.

Correlation function popupmenu: The thresholds can be displayed and edited in a correlation-function specific manner.

Image to display popupmenu: By default, the G(1)-G(end) images are displayed, but the different above images can be displayed via the popupmenu.

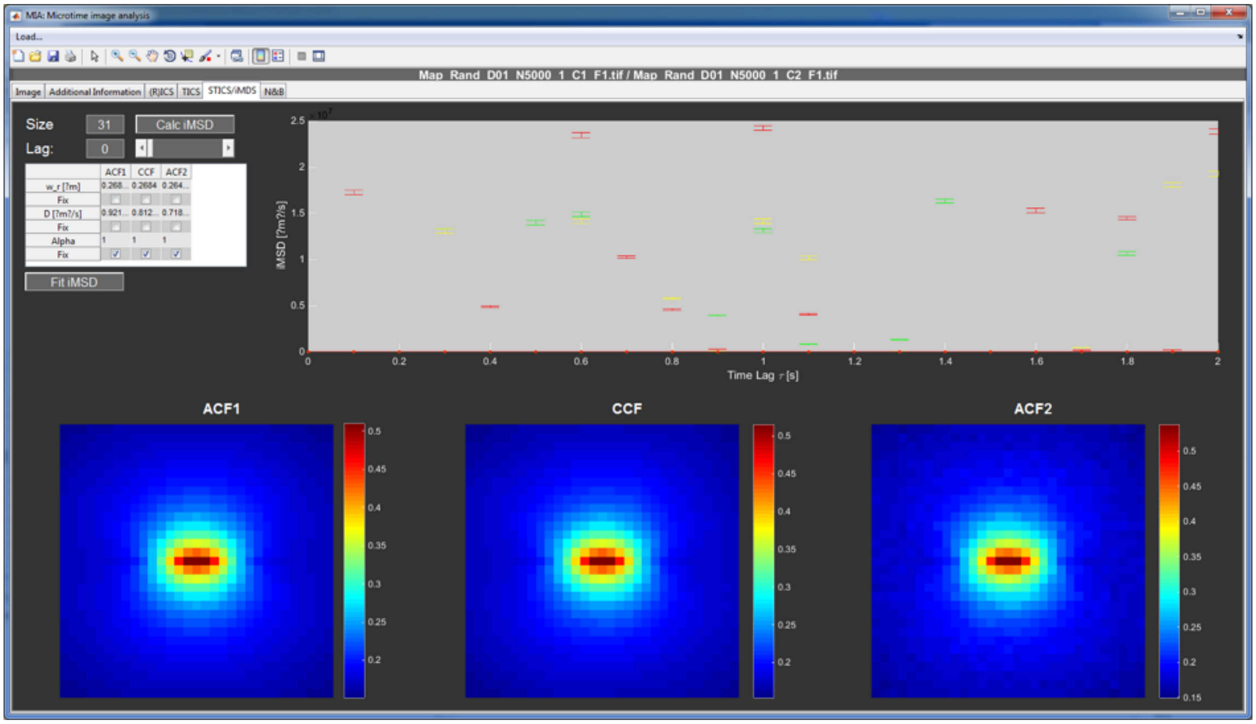
Median filter: median filters the resulting images (Radius defined on the **Correlate** tab)

Normalize: normalizes the display of the 1D average TICS correlation graphs between 1 and 0, for better displaying.

Reset: resets the thresholds.

5.6 STICS/iMSD tab

MIA can not only perform spatial or temporal correlation but also spatiotemporal correlation. The full STICS data can be properly analyzed with **MIAFit** when saved as a MATLAB file with the *.stcor* extension. In the **STICS/iMSD** tab the user can look at the data as a function of lag time in the three images at the bottom (left and right for the auto- and the central for the crosscorrelation).



Size 31

Lag: 0

	ACF1	CCF	ACF2
w_r [?m]	0.268...	0.2684	0.264...
Fix	□	□	□
D [?m?/s]	0.921...	0.812...	0.718...
Fix	□	□	□
Alpha	1	1	1
Fix	☑	☑	☑

Fig. 12: The STICS tab

A simplified way of analyzing spatio-temporal correlation data is called iMSD. Here, the square of the width of the Gaussian used to fit the correlation is plotted vs. the time lag. This way of plotting the data contains in principle the same information as mean-square displacement plots acquired by tracking. The iMSD data can be calculated, plotted and fit with the controls at the top of the window. For a more detailed analysis, the user can select to save the iMSD data during correlation as a MATLAB file with the *.mcor* extension. Since iMSD gives 1D temporal data, it can be analyzed with an appropriate fit in **FCSFit**.

5.7 Number and Brightness tab

N&B analysis is still under development, so use with caution!

The absolute number and brightness are calculated in Mia from absolute photon counting images using Eqs. 8 and 9 from [Digman2008], with the difference that a one-photon PSF shape correction factor = sqrt(8) is included (Eqs. 16-17 in [Hendrix2015]) to be consistent with RICS analysis.

For analog or pseudo photon counting (e.g. Olympus Fluoview FV1000) data, instead of using Eqs. 8 and 9 from [Dalal2008], in Mia images are *converted to absolute photon counting* after which Eqs. 8 and 9 from [Digman2008] can still be used.

Data correction: The N&B program uses the formulas for absolute photon counting A procedure for correcting data prior to calculating N&B can be found in [Hendrix2015]. In brief, for N&B, all slow fluctuations that are not caused by diffusing molecules need to be removed, but, in contrast to RICS, spatial image inhomogeneities due to e.g. concentration differences do not need to be removed. Ideally, subtracting a pixel's moving average followed by adding the pixel's average throughout the video would work. In our experience, however, we find it better to subtract the moving average of the 'local spatial average of a pixel', followed by adding the pixel's average throughout the video. Therefore, on the Correction tab in Mia, select

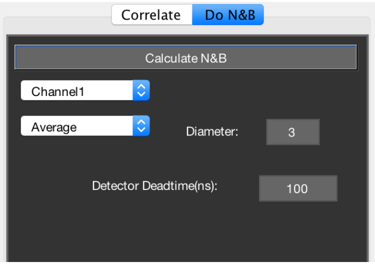
- subtract *moving average*: *pixels* = 3, *frames* = 3

- add *pixel mean*

Notes:

- Since N&B works on a pixel-by-pixel level, do not select an *arbitrary region ROI*.
- Any image *background correction* that you might do will ensure the N&B data is background corrected too.

N&B calculation: On the *do N&B* tab in Mia, the following options are visible:



Correlate | Do N&B

Calculate N&B

Channel1

Average

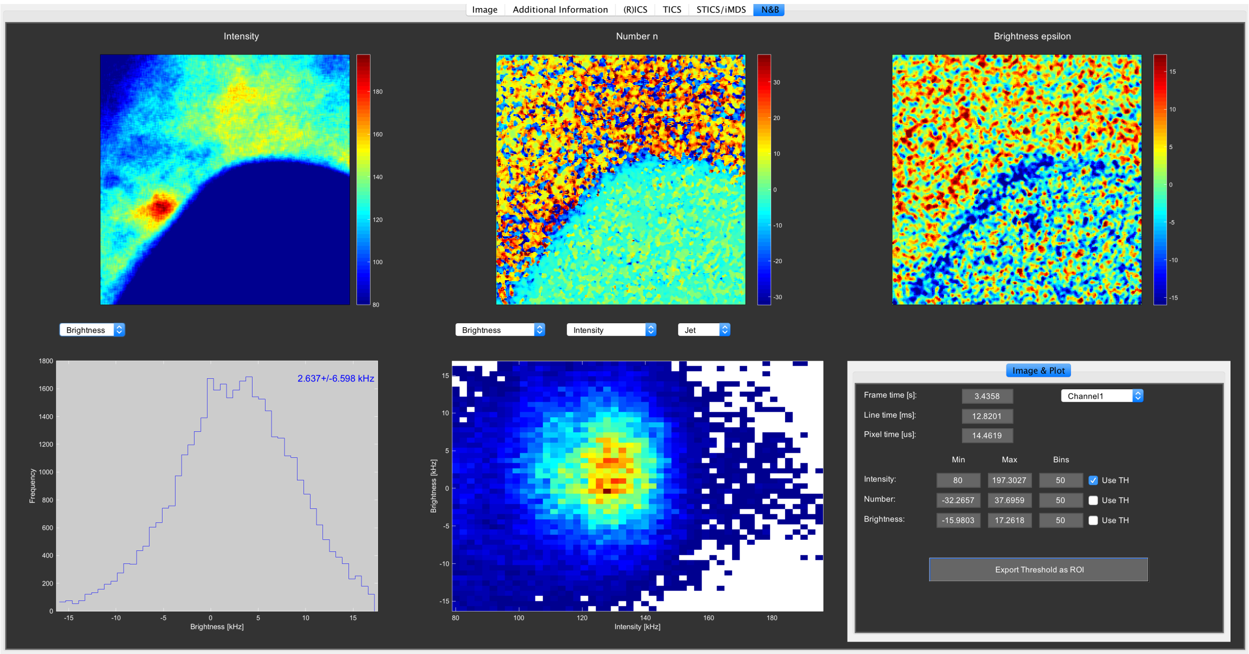
Diameter: 3

Detector Deadtime(ns): 100

Fig. 13: The N&B calculation tab

- *Channel 1* popupmenu: select which data channel you want to use for normal N&B [Digman2008], or whether you want to do cross-NB [Digman2009b].
- *Average* popupmenu: For N&B, the program will calculate the mean and variance image. This option allows you to additionally spatially average ('smooth') these images prior to calculating the N&B images.
- *Radius*: select the smoothing filter radius in pixels.
- *Median filter* checkbox: applies a [radius x radius] median filter to the N&B images.
- *Detector deadtime (ns)*: as described in [Hendrix2013], detector/TCSPC dead time will cause artifacts in N&B due to the fact that the highest pixel counts are systematically missing some counts, skewing the PCH and thus the N&B data. If you don't use TCSPC, a value of 0 will probably be ok. If you use TCSPC, look up the device's dead time. APDs also have a dead time, so take whichever dead time is highest (TCSPC or APD). Hybrid and normal PMTs have very low dead time (~1 ns). If you're unsure about dead time artifacts, compare the N&B brightness of different concentrations of the same sample (e.g. 10, 50, 250, 1000 kHz). If the brightness artificially goes down, try correcting the dead time to see if the effect disappears.

N&B data display: On the *N&B* tab, the calculated N&B data can be displayed and manipulated:



Frame time [s]: 3.4358 Channel1
Line time [ms]: 12.8201
Pixel time [us]: 14.4619

	Min	Max	Bins	
Intensity:	80	197.3027	50	☑ Use TH
Number:	-32.2657	37.6959	50	☐ Use TH
Brightness:	-15.9803	17.2618	50	☐ Use TH

Fig. 14: The N&B tab in Mia

- *Intensity, number and brightness* images display the corresponding parameters in each of the images.
- *Intensity, number, brightness and PCH* histogram displays the corresponding informations.
- The *2D histogram* allows plotting different parameters against each other.
- You can set the thresholds bottom right on the user interface to make the display of the images and histogram look better.
- Clicking the 2D histogram changes the background from white to black
- You can Export the applied thresholds as a mask that can be opened again on the ROI tab in Mia.
- Note: for the correct conversion of pixel counts to kHz, the correct pixel dwell time is needed!

General notes on N&B:

- For a dataset of 100 frames, try to have brightnesses at least >5 kHz, otherwise the data is too noisy for decent N&B.
- Increasing the moving average or filter averaging radi will improve data quality.

5.7.1 Converting analog to photon counting data

For determining absolute brightnesses (kHz) with e.g. *RICS* and *N&B* using imaging data recorded in analog mode, and for determining absolute number concentrations with N&B using analog data, the images need to be converted: (Image-offset)/S, with the parameters defined in [Dalal2008]. Here is a brief protocol on how to record the parameters.

- Put a drop of MQ on a coverslip, or MQ in the wells you use for measuring your cells.
- You measure 20 frames with your normal RICS settings 256x256, pixel dwell time 8-20 µs, pixel size ~50nm, BUT instead of using your laser ("unclick" to use the laser), you are creative with light that has different intensity.
- Use roomlight / no light / cellphone light / a bike lamp / . . . shining in various directions towards your sample. Try to obtain a nice spread in lamp intensity.

- For focussing, (if needed, put something in the well next to it) turn on the laser, 10x higher than usual, put the LUT to see dim signals, and you should find the position where the laser is reflecting the glass, go ~10 µm upwards.
- For analysis, open the file in Mia and simply copy paste the following code to the Matlab command:

```
global MIAData

disp(['there are ' num2str(size(find(MIAData.Data{1}>255),1)/numel(MIAData.
↪Data{1})*100) ' % saturated pixels']);

int = mean(mean(mean(double(MIAData.Data{1}),3)))

var = mean(mean(var(double(MIAData.Data{1}),0,3)))
```

- make a variance vs. intensity plot

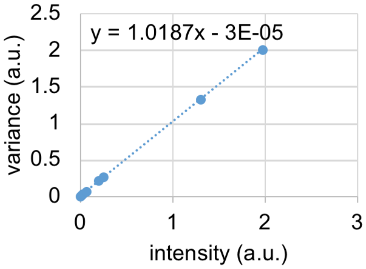


Fig. 15: Variance vs. Intensity plot

- Fit a straight line to the data. Slope = S, intercept = offset. Enter those values into Mia. As the data is immediately converted, all downstream calculations are now done on photon counting data.
- Importantly, RICS and N&B analysis should now provide exactly the same number and brightness.

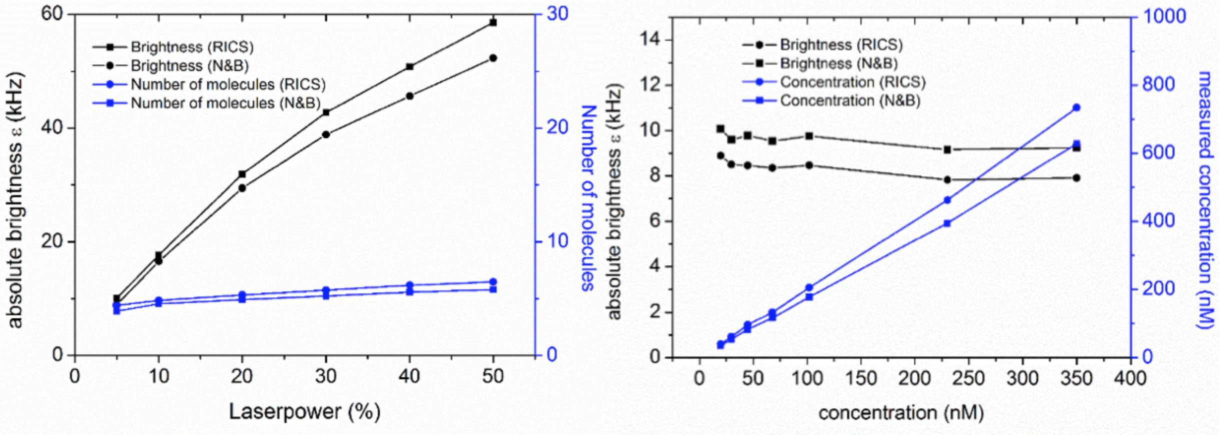


Fig. 16: Comparison of RICS and N&B using analog => photon counting converted data (Olympus Fluoview FV1000)

5.8 Spectral RICS

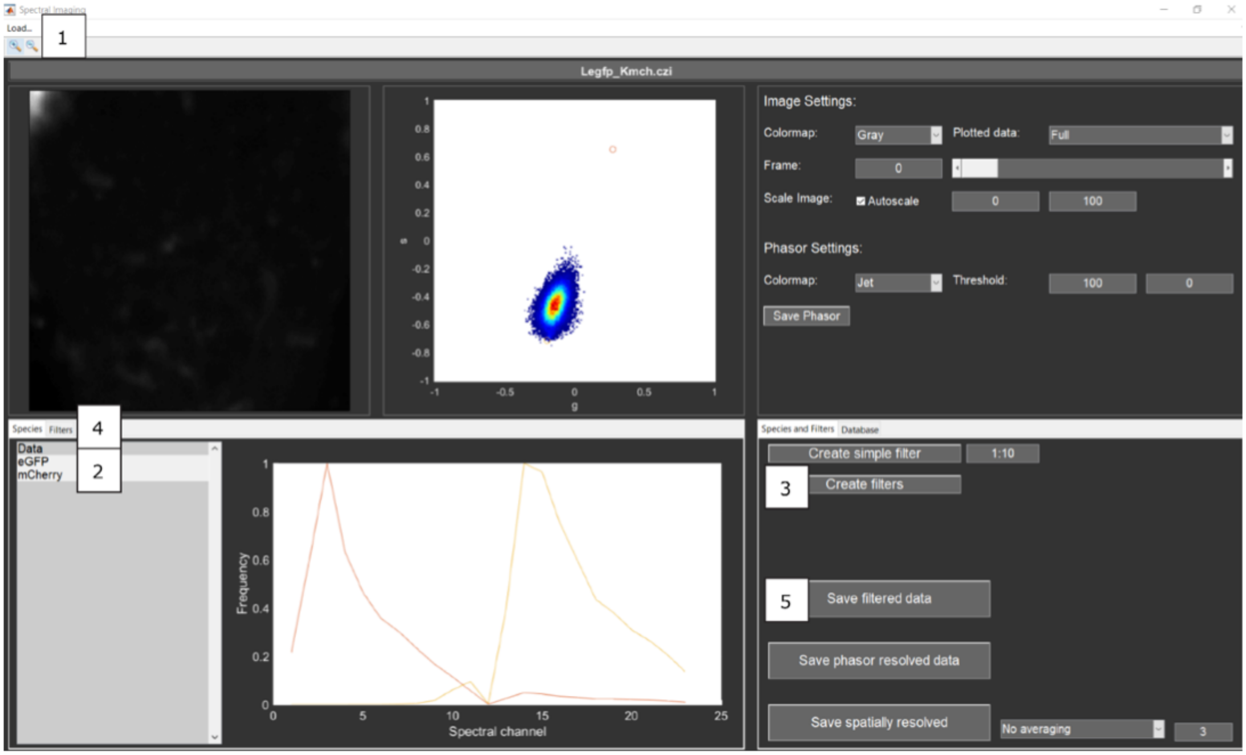
From the menu **Open…** **Spectral** a user interface can be called that allows generating spectral filters and applying these to RICS data. Alternatively, the Spectral UI can be opened from the command window by typing "Spectral".

Data can be loaded one by one via **Load -> data directly** (highlight #1).

For spectral filtering of data, single-species reference data can be loaded via **Load -> species data** (highlight #1). When selected (highlight #2), the relative frequency of photons detected in the different channels can be observed in the graph.

With the mixed data and pure-species data loaded, by pressing the **Create filters** pushbutton when both reference species are still selected (highlight #3), filter patterns are created.

In order to process the image series with the spectral filter, go to the **Filters** tab (highlight #4).



Colormap: Gray
Plotted data: Full
Frame: 0
Scale Image: ☑ Autoscale 0 100
Colormap: Jet
Threshold: 100 0

Fig. 17: The Spectral User Interface

Select the filters on the **Filters** tab by pressing the "+" or "right-arrow" key (highlight #4) (the filter's name will turn green) and pressing the **Save filtered data** pushbutton (highlight #5). This action creates two .tif files, one weighted towards the green and one towards the red species. Specifically, the signal intensity of each spectral bin of each pixel is multiplied with its corresponding weight, and every weighted pixel is summed up over all spectral bins [Schrimpf2018].

Batch processing is possible via load -> data to database.

The newly generated .tif files contain both the weigthed and unweighted data. They can each be opened and further analysed fr

Choose the **…weighted TIFFs** option to select the .tif files per channel. This will display the spectrally weighted data. Mind that the resulting images contain both positive and negative values, this is normal. Choose the **. . . single color TIFFs** option to select the .tif files per channel. This displays simply the sum of all spectral channels, and is not so useful.

The original .czi files contain the spectral imaging data. Load the preferred sum of spectral channels into Mia by defining the sp

Choose the **. . . Custom data format** option to select the original .czi file.

5.9 RICS Performance Evaluation

From the menu **Open. . . RICSPE** a user interface can be called that allows predicting the optimal RICS image acquisitioning parameters given particular sample properties (concentration, brightness, diffusion constant) [Longfils2019].

5.10 References

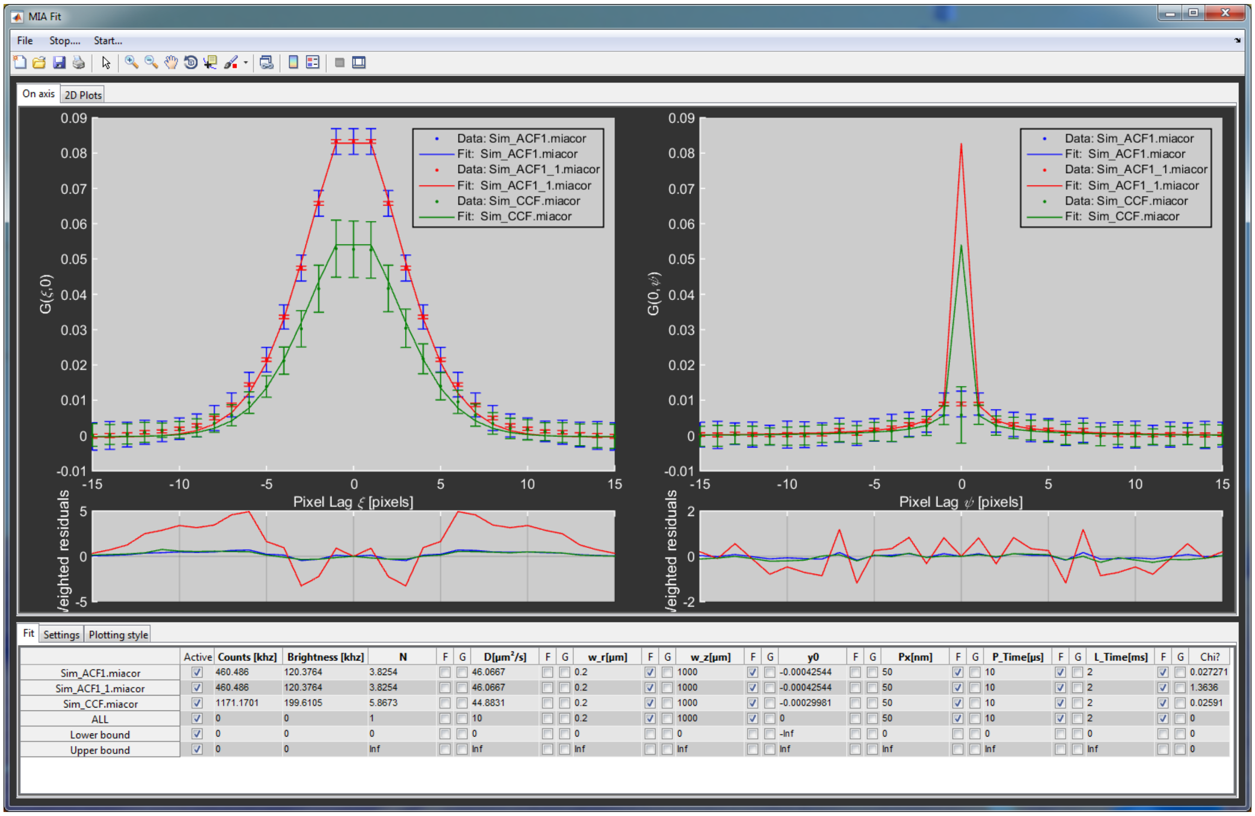
CHAPTER 6

MIAFit

Contents

Fitting of 2D and 3D correlation data can be accessed from the **PAM** main window by clicking on **Advanced Analysis -> MIAFit** or from the command window by typing in **MIAFit**.

The program work in many ways identical to *FCSFit* with small differences in displaying the data. The upper part of the window is dedicated for displaying the correlation data, either as 1D on-axis plots for all files, or as a 2D representation for a single file. In the lower part of the window the user can set the fit parameters and other fitting and display settings.



	Active	Counts [khz]	Brightness [khz]	N	F	G	D[µm²/s]	F	G	w_r[µm]	F	G	w_z[µm]	F	G	y0	F	G	Px[nm]	F	G	P_Time[µs]	F	G	L_Time[ms]	F	G	Chi?
Sim_ACF1.miacor	☑	460.486	120.3764	3.8254	☐	☐	46.0667	☐	☐	0.2	☑	☐	1000	☑	☐	-0.00042544	☐	☐	50	☑	☐	10	☑	☐	2	☑	☐	0.027271
Sim_ACF1_1.miacor	☑	460.486	120.3764	3.8254	☐	☐	46.0667	☐	☐	0.2	☑	☐	1000	☑	☐	-0.00042544	☐	☐	50	☑	☐	10	☑	☐	2	☑	☐	1.3636
Sim_CCF.miacor	☑	1171.1701	199.6105	5.8673	☐	☐	44.8831	☐	☐	0.2	☑	☐	1000	☑	☐	-0.00029981	☐	☐	50	☑	☐	10	☑	☐	2	☑	☐	0.02591
ALL	☑	0	0	1	☐	☐	10	☐	☐	0.2	☑	☐	1000	☑	☐	0	☐	☐	50	☑	☐	10	☑	☐	2	☑	☐	0
Lower bound	☑	0	0	0	☐	☐	0	☐	☐	0	☐	☐	0	☐	☐	-Inf	☐	☐	0	☐	☐	0	☐	☐	0	☐	☐	0
Upper bound	☑	0	0	Inf	☐	☐	Inf	☐	☐	Inf	☐	☐	Inf	☐	☐	Inf	☐	☐	Inf	☐	☐	Inf	☐	☐	Inf	☐	☐	0

Fig. 1: The main GUI of MIAFit.

6.1 Data handling and loading data

Currently, the program only supports files created with **PAM**, both based on MATLAB files (*.miacor*) and TIFF files with additional information added in an info file. A third type are STICS files (*.stcor*) that treat the different lags as individual files.

To load new files, got to **File -> Load New Files** or **File -> Add Files**. The former clears previously loaded files and load the new ones, while the later adds new files to the loaded ones. All loaded files are listed in the table and the bottom. Individual files can be removed by clicking on the **Remove** checkbox in the **Plotting Style** tab. Only one STICS file can be loaded at once and all previous files will be cleared.

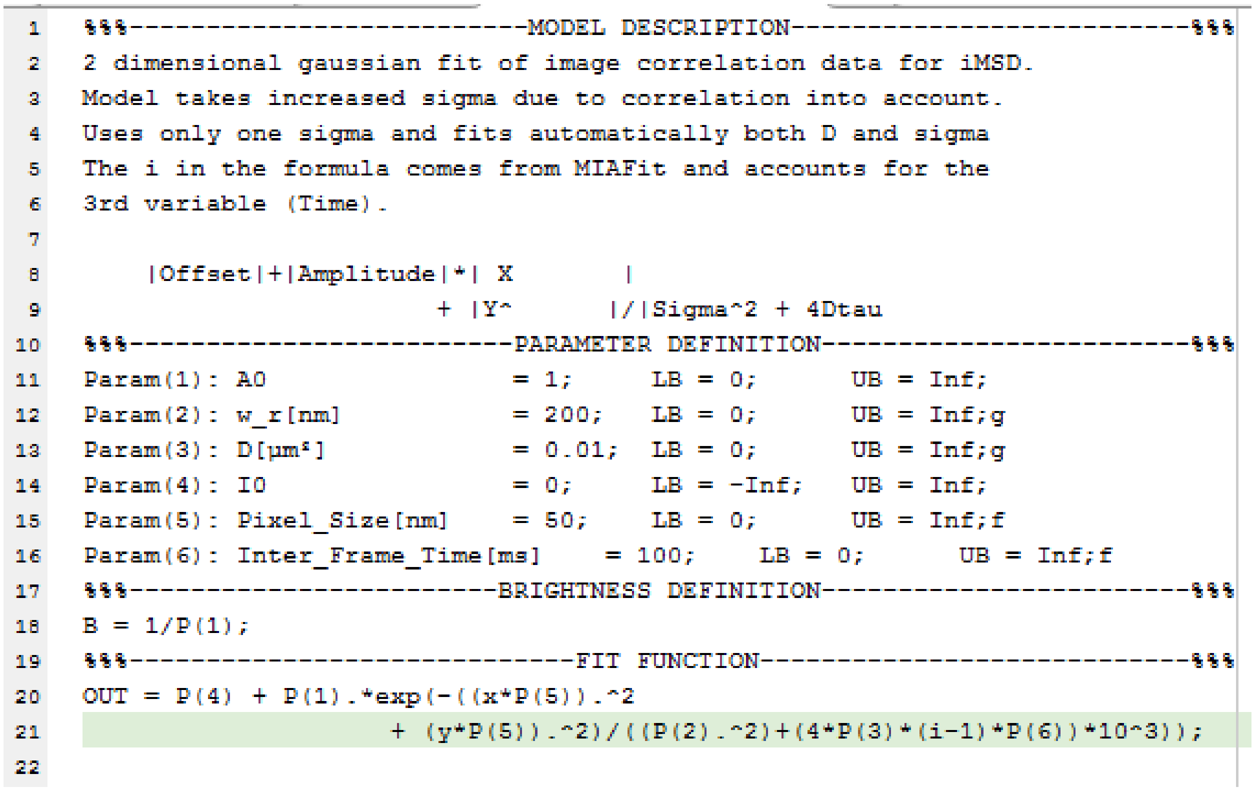
The raw correlation data is accessible from the command line via the global variable *MIAFitData*, and the fits and other processed parameters are stored in the global variable *MIAFitMeta*.

6.2 Fitting models and algorithms

Fitting in **MIAFit** is not based on hard coded fit models but instead uses models stored in text files (*.miafit*). This allows users to easily create new models and modify old ones without having to change the code itself, especially useful for a compiled program.

6.2.1 Fit models

MIAFit uses text based fit models. An example of the structure of a model is shown here:



```
%%%--------------------------MODEL DESCRIPTION--------------------------%%%
2 dimensional gaussian fit of image correlation data for iMSD.
Model takes increased sigma due to correlation into account.
Uses only one sigma and fits automatically both D and sigma
The i in the formula comes from MIAFit and accounts for the
3rd variable (Time).

    |Offset|+|Amplitude|*| X        |
                        + |Y^      |/|Sigma^2 + 4Dtau
%%%-------------------------PARAMETER DEFINITION------------------------%%%
Param(1): A0                = 1;     LB = 0;      UB = Inf;
Param(2): w_r[nm]           = 200;   LB = 0;      UB = Inf;g
Param(3): D[µm²]            = 0.01;  LB = 0;      UB = Inf;g
Param(4): I0                = 0;     LB = -Inf;   UB = Inf;
Param(5): Pixel_Size[nm]    = 50;    LB = 0;      UB = Inf;f
Param(6): Inter_Frame_Time[ms]    = 100;     LB = 0;      UB = Inf;f
%%%------------------------BRIGHTNESS DEFINITION------------------------%%%
B = 1/P(1);
%%%------------------------------FIT FUNCTION---------------------------%%%
OUT = P(4) + P(1).*exp(-((x*P(5)).^2
                    + (y*P(5)).^2)/((P(2).^2)+(4*P(3)*(i-1)*P(6))*10^3));

```

Fig. 2: An example fit model for MIAFit.

Each model file consists of four segments. The headers of the segments are needed for the program to recognize the individual parts and should not be modified.

Model description: This part is not used by the program itself but is used as a description for the user to understand the purpose and outline of the fit.

Parameter definition: In this section the different parameters used for the fit are defined and initial values set. The nth parameter is defined by *Param(n):*, followed by the name of the parameter displayed in **MIAFit** (do not use spaces in the name as they are used as separators). The next settings in order are the initial value, the lower and the upper limits. At the end you can use *g* or *f* to the set the parameter to global or fixed during initialization of the fit. The equal signs and semicolons are essential as the program searches for them as separators between the entries.

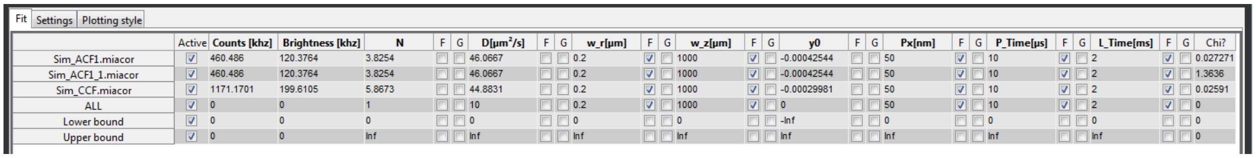
Brightness definition: The program uses this part to calculate the molecular brightness. Generally, the brightness is defined as *1/N*, but can be more complex for other models. Use standard MATLAB nomenclature to define this function and terminate it with a semicolon.

Fit function: Here the actual function used for fitting is defined. The individual parameters are abbreviated as *P(n)* and the variables are *x* and *y*. The additional variable *i* can be used for 3D correlation data. This value is given by the file number and is mostly used for STICS data. Keep in mind that the first file of *.stcor* data (i.e. *i=1*) contains the zero lag correlation. Use standard MATLAB nomenclature to define the function. Multiline entries are possible. In such a case use simple line breaks and not the `...` used in MATLAB scripts and functions.

6.2.2 Fitting procedure

When starting **MIAFit**, the model used during the last session is pre-loaded. A different fit function can be selected via **File -> Load Fit Function**. The standard folder for this is the *models* sub-folder of **PAM**.

The file names and the fit parameters defined in the model are shown in the table in the **Fit** tab.



	Active	Counts [khz]	Brightness [khz]	N	F	G	D[µm²/s]	F	G	w_r[µm]	F	G	w_z[µm]	F	G	y0	F	G	Px[nm]	F	G	P_Time[µs]	F	G	L_Time[ms]	F	G	Chi?
Sim_ACF1.miacor	☑	460.486	120.3764	3.8254	□	□	46.0667	□	□	0.2	☑	□	1000	☑	□	-0.00042544	□	□	50	☑	□	10	☑	□	2	☑	□	0.027271
Sim_ACF1_1.miacor	☑	460.486	120.3764	3.8254	□	□	46.0667	□	□	0.2	☑	□	1000	☑	□	-0.00042544	□	□	50	☑	□	10	☑	□	2	☑	□	1.3636
Sim_CCF.miacor	☑	1171.1701	199.6105	5.8673	□	□	44.8831	□	□	0.2	☑	□	1000	☑	□	-0.00029981	□	□	50	☑	□	10	☑	□	2	☑	□	0.02591
ALL	☑	0	0	1	□	□	10	□	□	0.2	☑	□	1000	☑	□	0	□	□	50	☑	□	10	☑	□	2	☑	□	0
Lower bound	☑	0	0	0	□	□	0	□	□	0	□	□	0	□	□	-Inf	□	□	0	□	□	0	□	□	0	□	□	0
Upper bound	☑	0	0	Inf	□	□	Inf	□	□	Inf	□	□	Inf	□	□	Inf	□	□	Inf	□	□	Inf	□	□	Inf	□	□	0

Fig. 3: The fit parameter table of MIAFit.

The different rows correspond to the individual files. The last three rows are used to set parameters for all files, and the lower and the upper bounds, in order.

The first column after the filenames is used to (de-)activate individual files to include them in or exclude them from the fitting and displaying procedure. The following column shows the average count rate of the file. For cross-correlation it is the sum of both channels. This is followed by the molecular brightness, calculated from the brightness defined in the model time the count rate. The other columns correspond to the different fit parameters with three columns each. The first is an editbox that contains the actual value. The other two are checkboxes used to either fix a parameter (F) or to globally link it between all files (G). The very last column represents the $\chi^2_{\text{red.}}$ of the fit.

For the actual fitting, **MIAFit** uses the standard MATLAB least-square curve fitting algorithm (*lsqcurvefit*) and linearizes the data. Only the data-points displayed in the upper plot (defined in the **Settings** tab) are used for fitting. When no variables are linked globally, each file is fit individually. The free parameters are used for the fit, while the fixed variables are given to the function as additional data to calculate the functions. For a global fit, the variables and fit-functions are concatenated and a single fit is executed. After the fit is finished, confidence intervals are calculated and the plots and tables updated. For auto-correlation curves, the (0,0) spatial lag point contains the noise peak. Furthermore, some cameras show strong artifacts for a line correlation. Therefore it is possible to omit either the center point of the whole line from the fitting algorithm in the **Settings** tab.

6.3 Data display

Displaying 2D data is much more complicated that for just a single dimension. **MIAFit** has therefore two different ways of visualization. The first shows the on-axis correlation of all loaded files with the corresponding error bars, fits and residuals. The left side shows the fast and the right side the slow axis for RICS data. The second display option shows the full correlation as a surface plot for a single file. Next to it, the user can display the fit, the residuals or a combination of both.

6.4 Additional setting and exporting

With the **Plotting Style** tab the user can change the appearance of the plotted data and fits. The **Setting** tab contains contains a variety of parameters for fitting and displaying, most notably the time range to be displayed and used for fitting (**Borders**), whether or not to use weights (**Use weights**), to omit the center point (**Center**), the whole y=0 line (**Line**), a specified number of lines (**Points**) and other fit related settings. Please be advised that data in which several points have to be omitted from the fitting might be compromised. Furthermore, it contains general setting for exporting the curves to a new MATLAB window. The curves can be exported to a new figure or the data can be exported to the MATLAB command line by right clicking the plot and selecting the appropriate function. Additionally, the **Copy**

Results to Clipboard button copies the fit parameters to the clipboard so they can be parsed into standard programs like Excel of Origin.

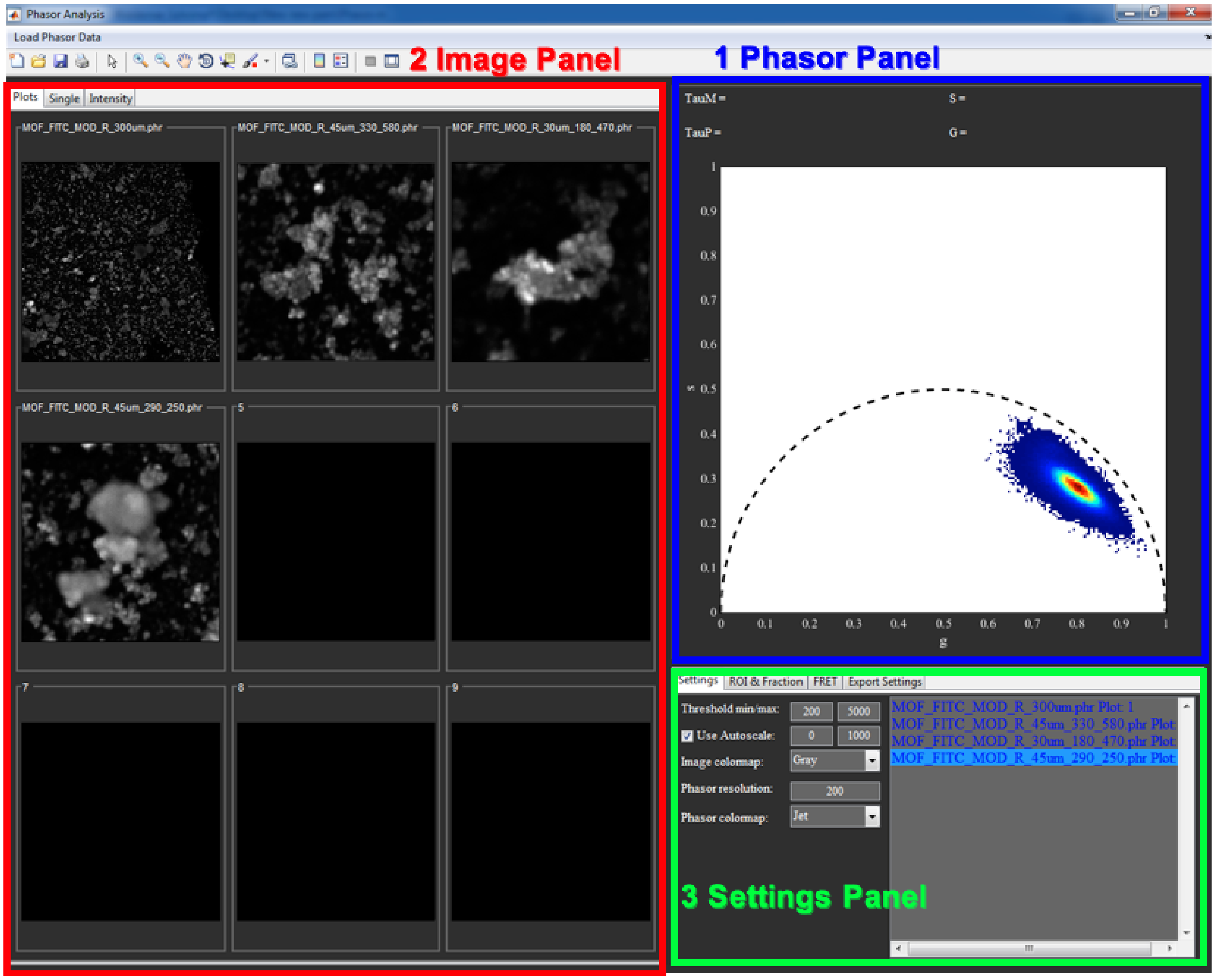
CHAPTER 7

Phasor Analysis

Contents

The phasor analysis window can be opened from the **PAM** main window by clicking on **Advanced Analysis -> Phasor** or from the command window by typing in `Phasor`.

The phasor approach to FLIM uses the Fourier space to analyze lifetime image data in a graphical way without the use of fit models, making it less biased compared to convolution or tail-fitting approaches. An introduction and description of the approach can be found in [Redfort2005] and [Digman2008]. An example application of the features of the Phasor analysis in PAM is featured in [Schrimpf2016].



Threshold min/max: 200 5000

☑ Use Autoscale: 0 1000

Image colormap: Gray

Phasor resolution: 200

Phasor colormap: Jet

MOF_FITC_MOD_R_300um.phr Plot: 1
MOF_FITC_MOD_R_45um_330_580.phr Plot:
MOF_FITC_MOD_R_30um_180_470.phr Plot:
MOF_FITC_MOD_R_45um_290_250.phr Plot:

Fig. 1: The main gui of Phasor.

The Phasor Analysis window is divided into three main panels:

1. **Phasor Panel:** Plots phasor histogram of activated files.
2. **Image Panel:**
    - Plots: Displays up to 9 images
    - Single: Larger representation of currently selected file
    - Intensity: Plots for Intensity vs. Frequency/G/S
3. **Setting Panel:**
    - Settings: File list and general display settings
    - ROI & Fraction: Display settings for ROIs and Fraction Line
    - FRET: Settings for FRET estimator
    - Export Setting: Controls for Export properties

7.1 General Data handling an plotting

7.1.1 Loading Data

To load new data, click on **Load Phasor Data** in the menu bar and select the files to be loaded. Currently the program only supports phasor files created with **PAM** (**.phr*). The selected files will be added to the **File List** in the **Settings** tab.

Phasor now also supports spectral phasor data with the ending **.phs*. Either spectral or lifetime data can be loaded at once. Loading the other type will clear previously loaded files.

7.1.2 Removing Data

To remove files from memory select them in the **File List** and press the ‘del’-Key.

7.1.3 Plotting multiple intensity images

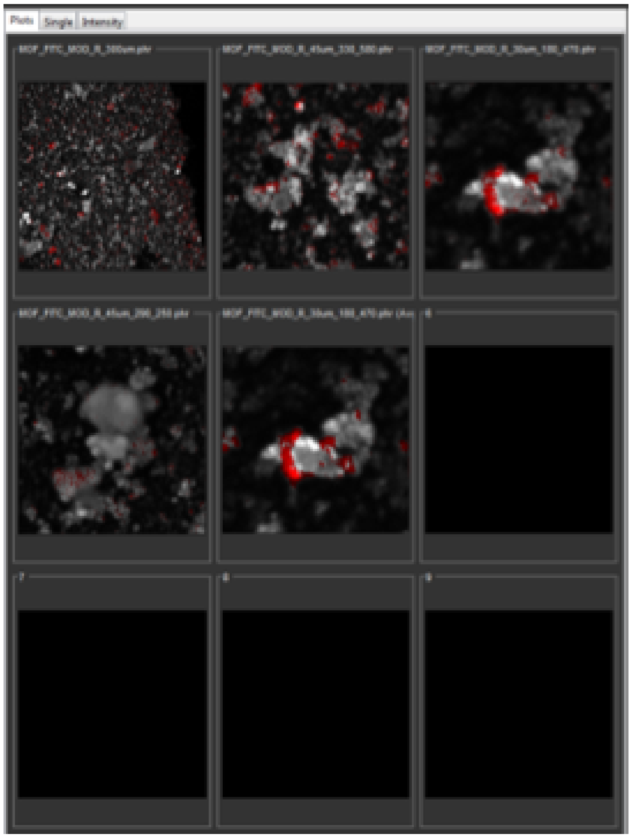
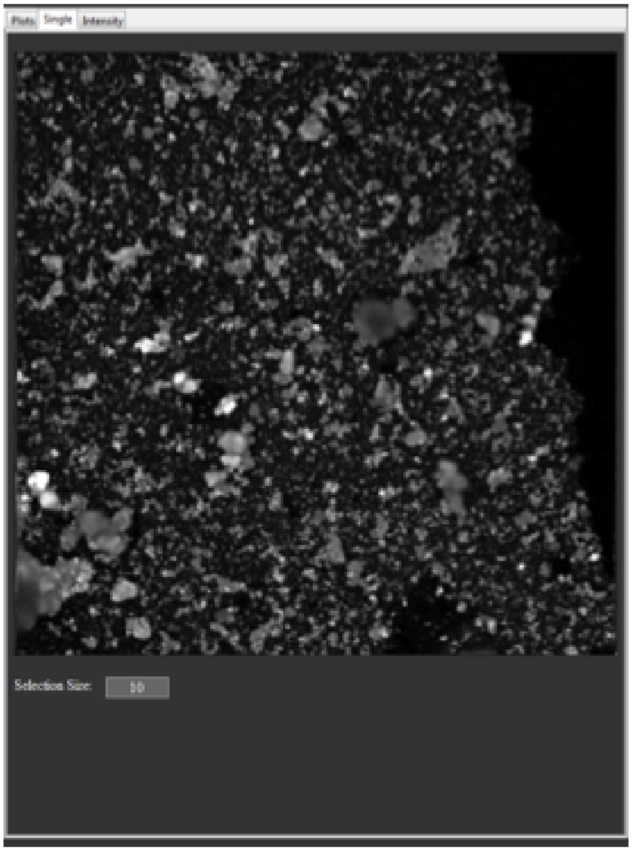

Plots
Single
Intensity
MOF_FITC_MOD_B_500um.phr
MOF_FITC_MOD_B_45um_330_580.phr
MOF_FITC_MOD_B_30um_180_470.phr
MOF_FITC_MOD_B_45um_290_250.phr
MOF_FITC_MOD_B_30um_180_470.phr (Au
6
7
8
9

Fig. 2: The Image panel.

A newly loaded file will be automatically plotted in the first empty plot in the **Plots** tab. Files currently displayed are indicated with a blue font and the number associated with the plot in the **File List**. To free up plots for other files, press the 'leftarrow'-Key while the **File List** is active. This deactivates the selected files, indicated by a white font. Deactivated files will not be plotted and not used for calculating the Phasor histogram. Pressing the '+'-Key will also remove the selected files from being plotted, but will keep them active. Active, but not plotted files are shown in a red font. Use the 'rightarrow'-Key to assign an unplotted file to a free image plot.

7.1.4 Plotting individual intensity images

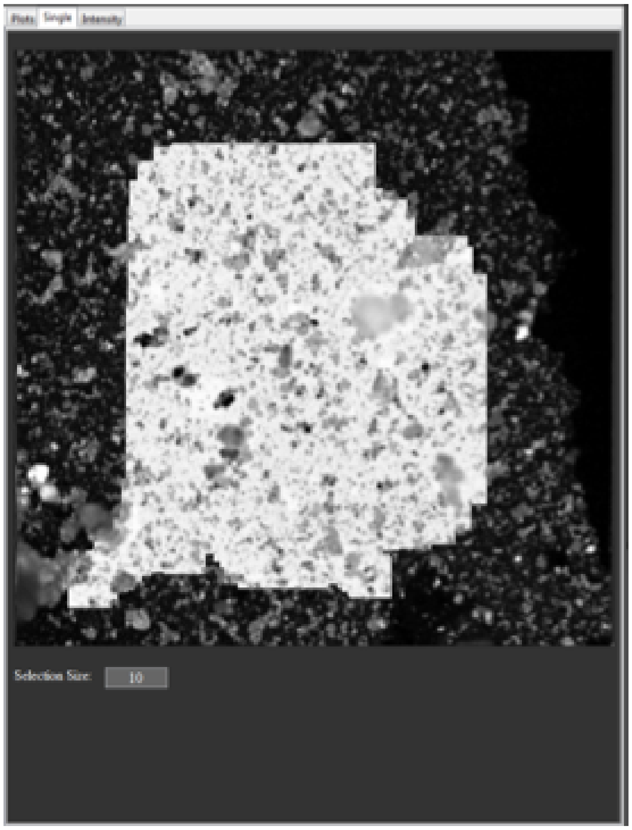


Selection Size: 10

Fig. 3: Plotting a single file.

To look at one individual file at a time in a bigger plot, select the **Single tab** in the **Image Panel**. Now, only the first selected file in the **File List** will be plotted and used to calculate the Phasor histogram, independent of whether it is active or not.

7.1.5 Plotting individual image sub-regions



Selection Size: 10

Fig. 4: Plotting an image subregion.

It is also possible to exclude pixels or regions of the image from the Phasor histogram. To do that, click and hold the *middle mouse button* in the **Single** plot. A square region around the cursor will be unselected and excluded when calculating the Phasor histogram (but only when in the **Single** tab). The size of the excluded region can be adjusted with **Selection Size**. To reselect individual regions, press and hold the left mouse button. Double clicking the **Single** plot will reselect the whole image. Clicking the *left mouse button* when all pixels are selected will unselect all pixels. Unselected regions are shown in inverted RGB colors (corresponding to [1 - R, 1 - G, 1 - B]).

7.1.6 Plotting the phasor center of mass

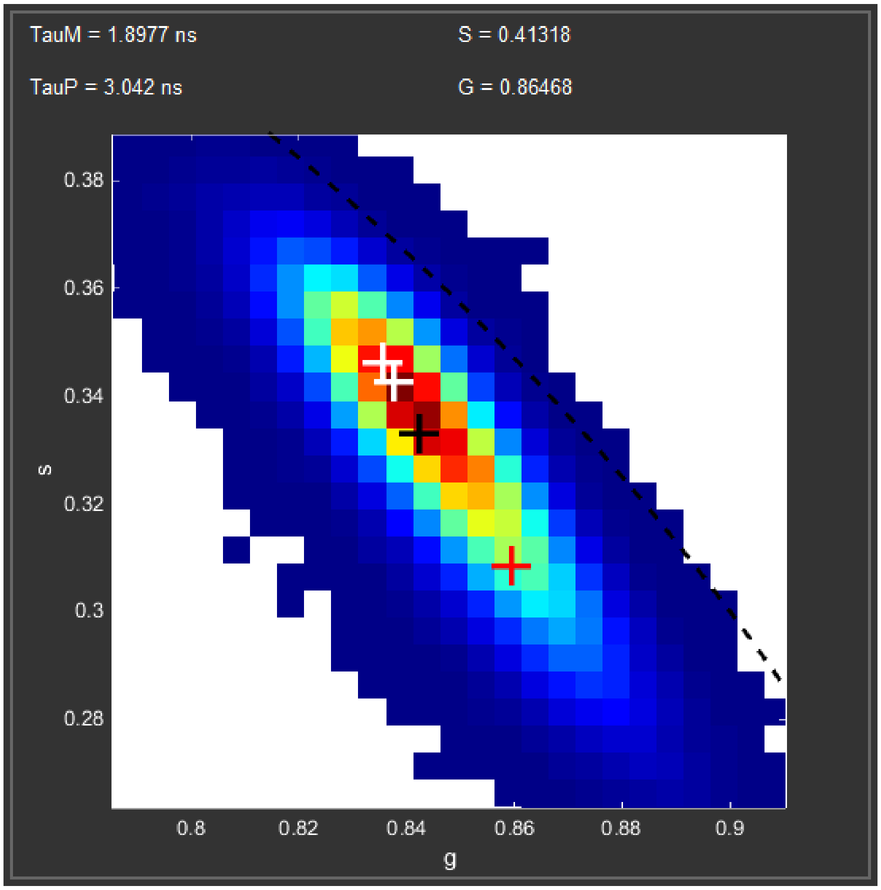


Fig. 5: Plotting center of mass.

To show the center of mass (CoM) of the individual files in the phasor histogram, select the option in the popupmenu in the bottom of the **Settings** tab. When either a pixel and photon weighted CoM display is selected, the program displays a marker indicating the CoM for each file used in the calculation of the phasor histogram. The calculation takes both the photon thresholds als well as the selected sub-region into account. To change the marker shape, use the popupmenu next to the **CoM Selection**. A *right click* on the shape menu opens a context menu with further options, like size or color of the marker. The marker properties will be applied to all files selected in the **File List**.

7.1.7 Changing image display styles

Intensity images are by default displayed using an autoscaled *Gray* colormap. To set the contrast manually, uncheck **Use Autoscale** in the **Settings** tab and type in lower and upper counts limits. Besides *Gray,* there are three standard Matlab colormaps *Jet*, *Hot* and *HSV*, as well as *Black*, a colormap that displays the image as black (sometimes useful in combination with ROIs). The different colormaps also behave differently when ROIs are selected, as is described in chapters 2.2, 2.4 and 2.6.

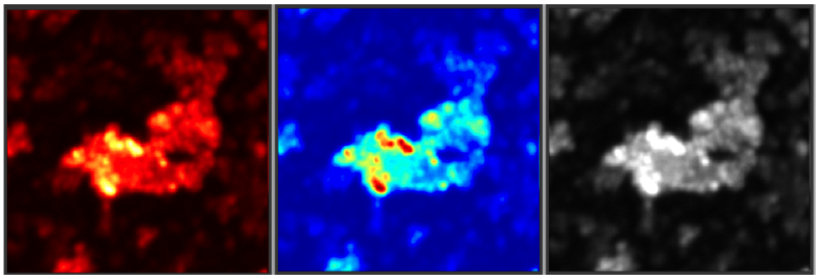


Fig. 6: Available colormaps.

7.1.8 Plotting Phasor Histogram

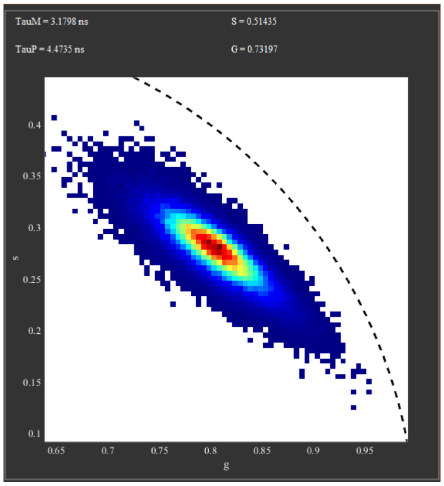


Fig. 7: The phasor plot.

The Phasor histogram is always calculated and plotted automatically. The black dotted line corresponds to the universal circle, highlighting all possible phasors for monoexponential decays. For spectral phasor data, the universal circle is remove and a circle with radius 1 around the origin is plotted to indicate all possible positions.

Depending on the selected tab in the **Image Panel** different data is used for calculation:

- **Plots:** All *active* files (blue or red font in **File List**) are used.
- **Single:** Only *selected regions* of first selected file in **File List** are used.
- **Intensity:** *All selected* files in **File List** are used.

In all cases all pixels outside the threshold (set via **Threshold min/max**) are discarded. Furthermore, values far outside the normal range (*g* or *s* <-0.1 or >1.2) are also not plotted. For spectral data this range is instead -1 to 1. The bin size of the Phasor plot can be adjusted using **Phasor resolution**. The value determines into how many bins the range 0 to 1 for both g and s is divided. Since the actual range is -0.1 to 1.2, the actual resolution is 1.3 times that value (**Caution**: The resolution must always be a multiple of 10, otherwise the program generates an error). As in the case of the images, different colormaps can be selected using **Phasor colormap**, with *Jet* as default.

7.1.9 Navigating in the Phasor Plot

If the cursor is inside the Phasor plot, the current position, as well as the lifetimes from Phase and Modulation are displayed on the top. By holding with the *left mouse button* and moving the mouse, one can reposition the plot, while the mouse wheel can be used to zoom in and out. The *middle mouse button* resets the default axis limits.

7.2 ROIs, Fraction Line and FRET

7.2.1 Selecting a Region of Interest (ROI)

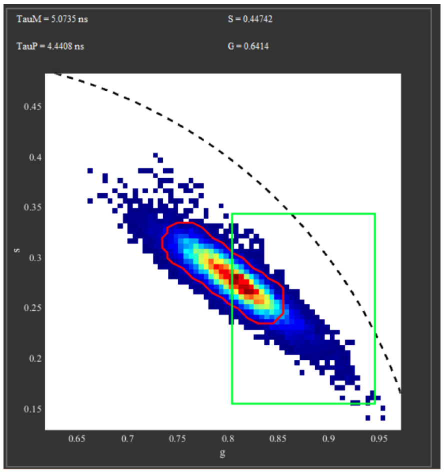


Fig. 8: Selecting ROIs in the Phasor plot.

It is possible to select and plot up to 6 different ROIs in the Phasor plot at the same time, each with one of two possible ROI shapes.

To select a ROI, press and hold keys 1-6 (Numpad also works). Next, click with the mouse inside the Phasor plot and select a ROI by moving the mouse, while keeping the mouse button pressed. If you use the left mouse button, the ROI will be a rectangle, while the right mouse button selects a roughly ellipsoid region. The diameter of the ellipsoid can be set for each ROI individually in the **ROI&Fraction** tab with the **Size** parameter.

There are two ways of unselecting ROIs: To completely delete a ROI, press and hold the number key and click once in the Phasor plot. The second method deactivates the ROI, while keeping its position in memory for later use. Clicking with the mouse on the respective **ROI Button** in the **ROI&Fraction** tab will toggle the corresponding ROIs. Hereby, left and right mouse buttons correspond to the rectangular and ellipsoid shapes, so that it is possible to keep both rectangular and ellipsoid ROIs.

7.2.2 Displaying ROIs

In the intensity plot, the color of pixels that have a phasor inside a ROI and that have an intensity within the set threshold, is changed to the color of the corresponding ROI. In case of overlapping ROIs, the resulting color is the average of the ROI colors.

The image colormap *Gray* comprises a special case. Here, the pixel color is the scaled intensity times the RGB values of the ROI color, therefore the brightness corresponds to pixel counts, while the ROI is indicated by the hue.

The line width and style of the ROI lines in the Phasor Plot can be changed in the **ROI&Fraction** tab with the **Width** and **Style** parameters, respectively. To assign a different color, click with the *middle mouse button* on the **ROI Button** and select a color from the menu.

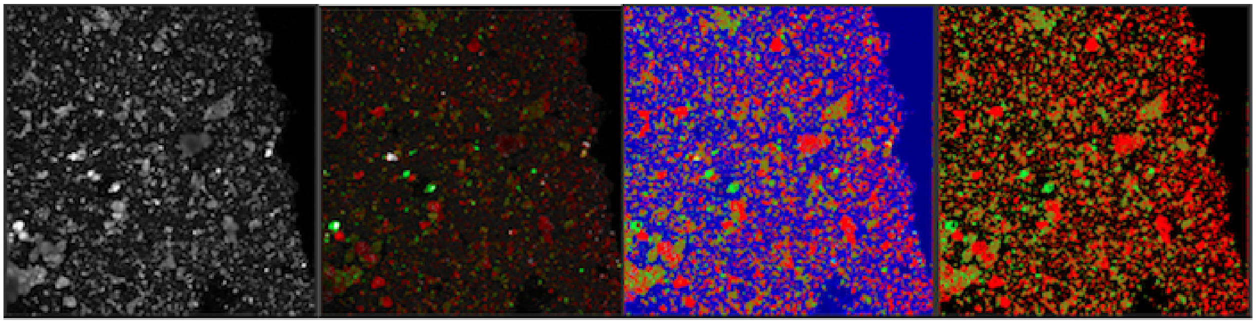


Fig. 9: Displaying phasor ROIs in the intensity images.

7.2.3 Selecting a Fraction Line

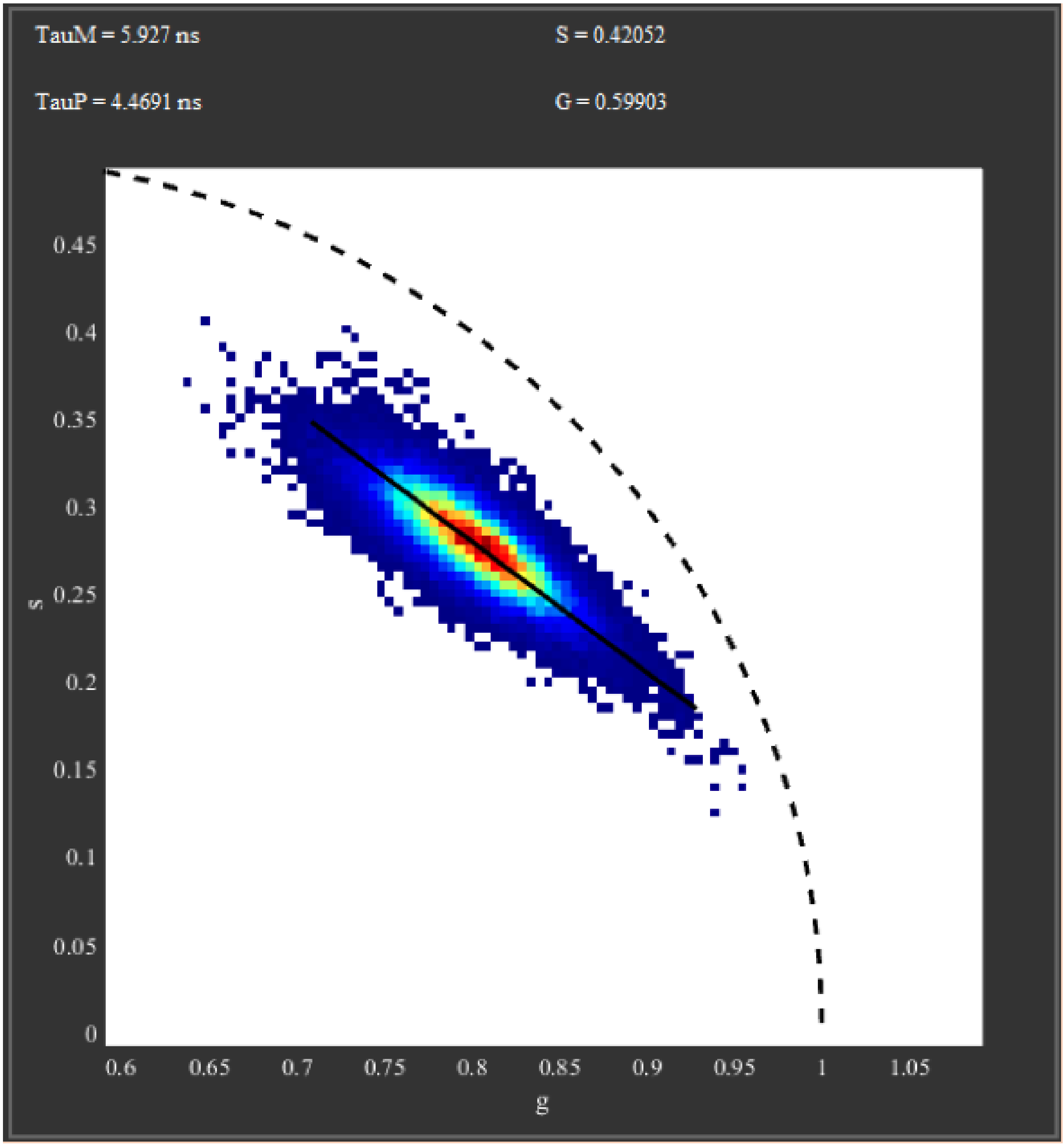


Fig. 10: Using the fraction line.

Since varying ratios of two components result in straight lines in a phasor plot, it is also possible to visualize these ratios with a straight line. Start selecting a Fraction Line by pressing and holding the '0'-Key (Numpad also works). Next, click with the mouse at the start position inside the Phasor plot and moving the mouse to the desired end position, while keeping the mouse button pressed. Clicking the *left* or *right mouse buttons* when selecting the **Fraction Line** will result in the reverse color order (explained in chapter 2.4). There are two ways of unselecting the **Fraction line**: To completely delete it, press and hold the 0-key and click once in the phasor plot. The second method deactivates the

Fraction Line, while keeping its position in memory for later use. Clicking with the mouse on the **Fraction Button** in the **ROI&Fraction** tab will toggle it.

7.2.4 Displaying a Fraction Line

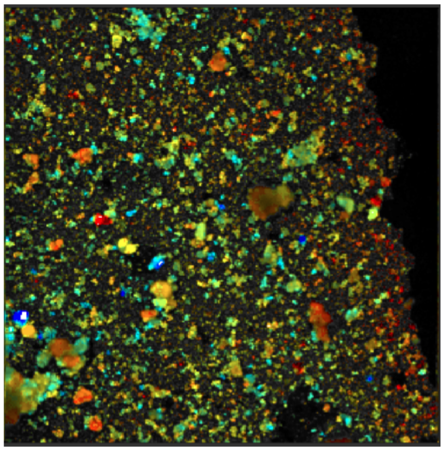


Fig. 11: Displaying the fraction line in the image.

After selecting a **Fraction Line**, pixels in the intensity plots will be color coded according to their position on the line. To be precise, the program determines for each pixel the closest point on the fraction line and assigns them a color corresponding to that point. Points further away than the Fraction **Size** set in the **ROI&Fraction** tab are not color coded. The colormap used for Fraction Line can be set with the popupmenu in the **ROI&Fraction** tab. The image colormap *Gray* comprises a special case. Here, the pixel color is the scaled intensity times the RGB values of the **Fraction Line** color, therefore the brightness corresponds to pixel counts, while the position in the line is indicated by the hue. The line width and style of the Fraction Line in the Phasor Plot can be changed in the **ROI&Fraction** tab with the **Width** and **Style** parameters, respectively. To assign a different color, click with the middle mouse button on the **Fraction Button** and select a color from the menu. Note that this color only affects the line plotted in the Phasor Plot, not the intensity images.

Hint: You can use custom colormaps to use for color-coding with the fraction line. To do so, create a colormap array (n x 3 double array of values between 0 and 1) in the matlab workspace and copy the data to the variable *Phasor_Colormap*. Then, declare *Phasor_Colormap* as a global variable. In the *Phasor* window, select *Custom* for the fraction line colormap. The program will then store the custom colormap for subsequent sessions.

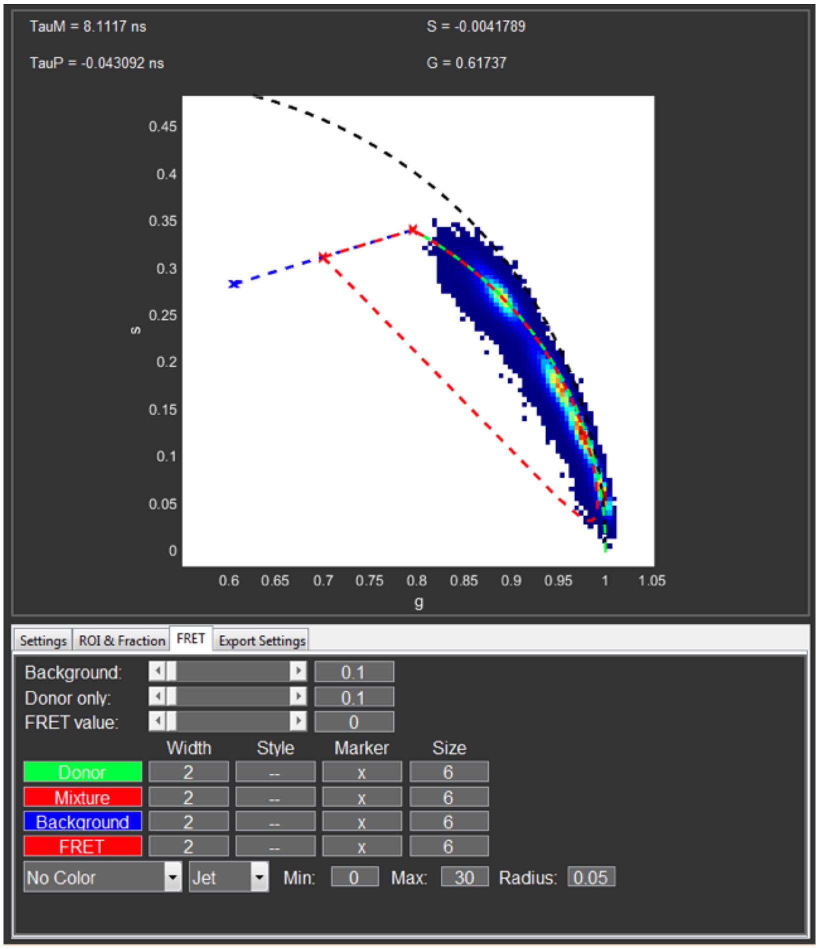
7.2.5 Selecting a Triangular Mixture

With a single phasor plot it is possible to unambiguously determine the fractions of a mixture of three components, as long as all three phasors of the base components are known. To display these mixtures, use the 7-, 8- and 9-keys to define a triangle. For this, hold down the corresponding key and click into the phasor histogram. The vertex of the triangle will then be placed at that position.

7.2.6 Displaying a Triangular Mixture

Once the triangle is defined, the program will color-code the images according to the pixels position in this triangle. For this, it calculates the relative fraction of each components and sets the red (7-key), green (8-key) and blue (9-key) color value to that fraction. As with the other display types, this RGB value can be adjusted with the pixels intensity for gray-scale displaying or as a flat value for all other colormaps.

7.2.7 Selecting FRET Trajectory



TauM = 8.1117 ns

S = -0.0041789

TauP = -0.043092 ns

G = 0.61737

Background: 0.1

Donor only: 0.1

FRET value: 0

	Width	Style	Marker	Size
Donor	2	--	x	6
Mixture	2	--	x	6
Background	2	--	x	6
FRET	2	--	x	6

No Color | Jet | Min: 0 | Max: 30 | Radius: 0.05

Fig. 12: FRET trajectory in phasor plot.

The phasor program is also able to calculate and display a curved quenching trajectory, mostly used for FRET. To start this type of selection, *right click* the phasor histogram at the position of the unquenched donor position and select **Set Donor**. The display will switch to the FRET selection mode and display several different lines and points. The green curve represents the ideal quenching curve of the pure donor (indicated by a green x) between 0 and 100% quenching. The blue line represents a mixture of the pure donor and a background contribution. The background phasor can be selected with *right click* the phasor histogram and selecting **Set Background**. The red curve represents the real FRET trajectory that takes into account the contribution of the background and a donor only species. These contributions can be adjusted with the corresponding sliders in the **FRET** tab. The display parameters can be changed in the **FRET** tab and the color can be changed with a *middle mouse button* click on the corresponding box.

7.2.8 Calculation of FRET Trajectory

For a mono-exponential decay, the phasor position as a function of quenching can be easily calculated. However, for a phasor inside of the universal circle no unique quenching trajectory exists, since the phasor can be a result of different combinations of lifetimes and fractions. Still, most of these combinations will have very similar trajectories. Therefore, the program assumes that the phasor is a result of a mixture of a mono-exponential decay τ and a contribution with a lifetime of 0. It then calculates the simple quenching for the mono-exponential decay τ and generates the trajectory by assuming a constant fraction between the two components (τ and 0). This pure FRET trajectory is plotted as a green curve. In reality, most measurement will not only contain the FRETing or quenched species, but also contributions from the background and unquenched donor molecules (due to missing of bleached acceptor dyes). The program accounts for that based on the values set by the user, resulting in the real FRET trajectory shown in red. Hereby, the contribution of background and unquenched donor are assumed to be constant. As quenching increases, the brightness of the FRETing species decreases, also decreasing its relative contribution to the photon stream. Therefore, the trajectory curves back towards the phasor of the constant contribution.

7.2.9 Displaying FRET Trajectory

The FRET trajectory can be displayed in two different ways. The first way works the same way as the fraction line, just with a curved line. To select this method of display, select **FRET Line** in the bottom left corner of the **FRET** tab. To increase the contrast, the relevant section of this line can be set with the **Min/Max** values. The colormap used here can be selected with the corresponding popupmenu.

The second way of highlighting the FRET trajectory is the **FRET Spot**. Here, a circular area in the phasor plot around the selected position of the real trajectory is highlighted in the images with the corresponding color (just like a ROI). The size of the spot, as well as the maximal distance from the FRET line can be set with the **Radius** value.

Hint: You can use costom colormaps to use for color-coding with the FRET line. To do so, create a colormap array (n x 3 double array of values between 0 and 1) in the matlab workspace and copy the data to the variable *Phasor_Colormap*. Then, declare *Phasor_Colormap* as a global variable. In the *Phasor* window, select *Custom* for the FRET line colormap. The program will then store the custom colormap for subsequent sessions.

7.3 Additional functions

7.3.1 Exporting Images

Any image that is currently plotted in the Phasor program can be exported to a new MATLAB figure or as a TIFF file by right clicking on the image and selecting the corresponding option. The images are exported as RGB data, so that the direct information about the pixels counts is lost.

7.3.2 Exporting Phasor Histograms

The phasor histogram, including all ROIs and other markings, can be exported as a separate MATLAB figure via a double click into the phasor histogram. The size and setting of the new figure can be set in the **Export Settings** tab.

7.3.3 Averaging Images

In some cases, the pixel counts in the image might be too low for an adequate analysis of the data. To improve the signal-to-noise the user can use a running average of 3x3 pixels on the phasor data. This can be via a right mouse button click in the **File List** in the **Settings** tab and selecting **Average Pixels**. This creates a new file that usually shows a narrower spread in the phasor histogram. However, this decreases the spatial resolution of the phasor data, but the resolution of the intensity image is maintained.

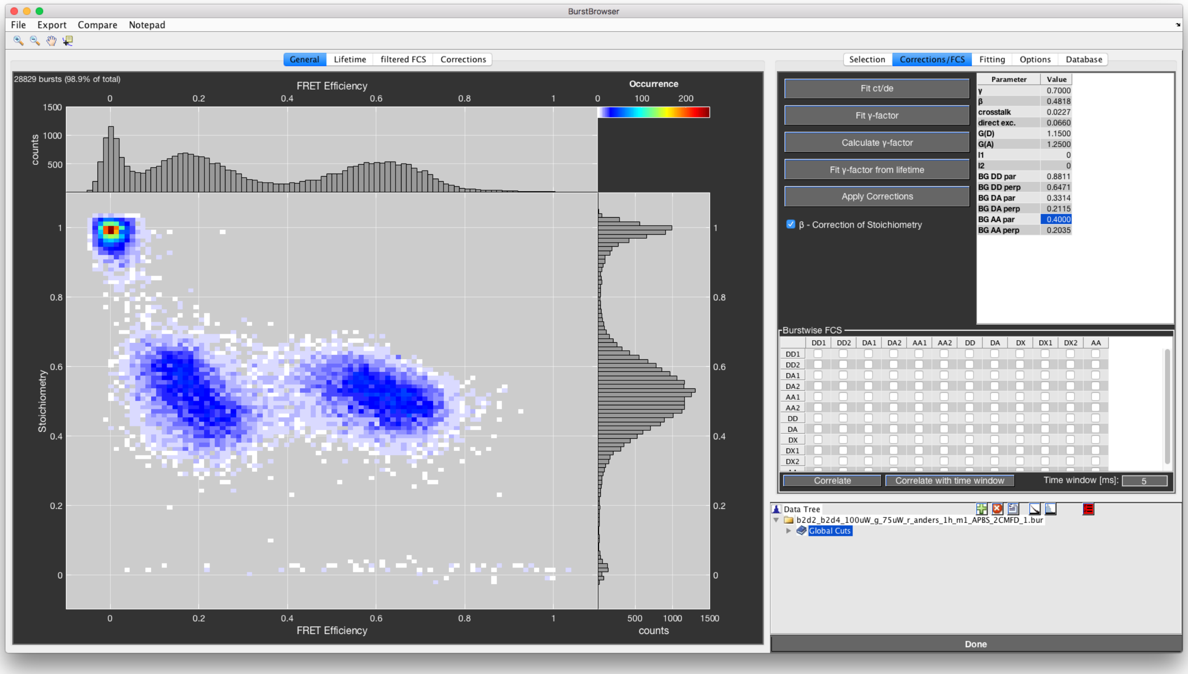
CHAPTER 8

BurstBrowser

Contents

8.1 Data exploration and plotting

Single-molecule FRET data can be interactively explored using BurstBrowser's main figure ("General" tab). Parameters are selected using the lists in the "Selection" tab. By default, BurstBrowser plots FRET efficiency vs. Stoichiometry.


BurstBrowser
File Export Compare Notepad
General Lifetime filtered FCS Corrections
28829 bursts (98.9% of total)
FRET Efficiency
Occurrence
0 100 200
counts
Stoichiometry
FRET Efficiency
counts
Selection Corrections/FCS Fitting Options Database
Fit ct/de
Fit γ-factor
Calculate γ-factor
Fit γ-factor from lifetime
Apply Corrections
☑ β - Correction of Stoichiometry
Burstwise FCS
Correlate
Correlate with time window
Time window [ms]: 5
Data Tree
b2d2_b2d4_100uW_g_75uW_r_anders_1h_m1_APBS_2CMFD_1.bur
Global Cuts
Done

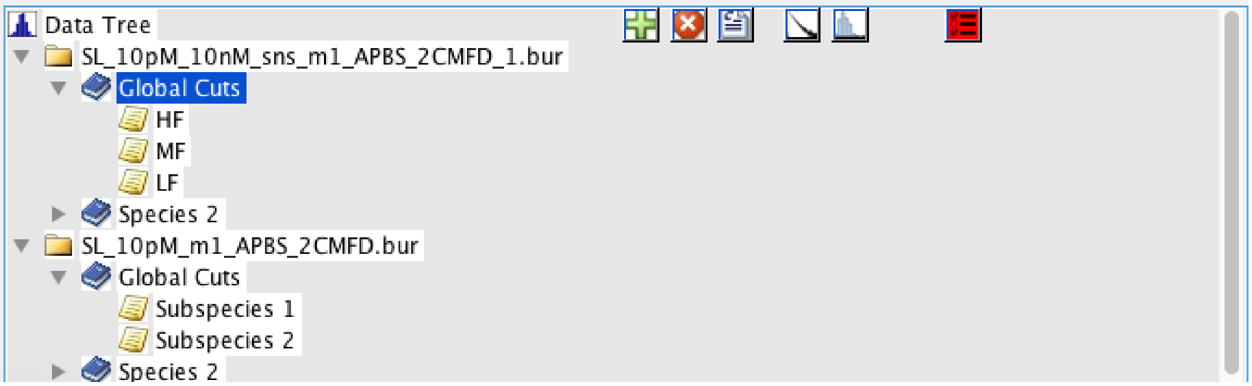
Parameter	Value
γ	0.7000
β	0.4818
crosstalk	0.0227
direct exc.	0.0660
G(D)	1.1500
G(A)	1.2500
l1	0
l2	0
BG DD par	0.8811
BG DD perp	0.6471
BG DA par	0.3314
BG DA perp	0.2115
BG AA par	0.4000
BG AA perp	0.2035

Fig. 1: BurstBrowser's main GUI.

8.1.1 Selecting data

To apply cuts to data, right-click the parameter to add it to the cut table. Here, you can set the lower and upper boundaries for the parameter ("min" and "max"). To temporarily disable the cut, uncheck the "active" checkbox. To remove it completely, click the "delete" checkbox. You can add any combination of parameters.

8.1.2 Using species


Data Tree

- SL_10pM_10nM_sns_m1_APBS_2CMFD_1.bur
  - Global Cuts
    - HF
    - MF
    - LF
  - Species 2
- SL_10pM_m1_APBS_2CMFD.bur
  - Global Cuts
    - Subspecies 1
    - Subspecies 2
  - Species 2

Fig. 2: The Species List

Oftentimes, it is helpful to sort the data into different populations or species, which are to be compared with respect to some parameters. BurstBrowser offers a convenient way to do so based on a species/subspecies hierarchy. Species are accessed using the “Data Tree” panel.

By default, BurstBrowser creates a single species called “Global Cuts”, with two subspecies (“Subspecies 1/2”), in addition to the top level tree element showing the filename. You can add species or subspecies by right-clicking an existing species and selecting “Add Species”, which will add an additional species on the same level. Species are removed by clicking “Remove Species”, and renamed by the menu item “Rename Species”.

The general idea of species is to first apply cuts to the measurement to filter out unwanted molecules (i.e. donor- and acceptor-only molecules, photobleaching events, multimolecule events), and subsequently refine the selection in terms of species (i.e. low FRET efficiency, high anisotropy, etc.).

The following rules apply:

- A newly generated subspecies inherits from the parent species.
- Cuts applied to a parent species will automatically be applied to all subspecies, meaning:
  - New parameters will be added to the subspecies.
  - Boundary changes will overwrite boundaries in the subspecies.
- Deleting a parameter from a parent species will remove it from all subspecies.
- Deleting a parameter in a subspecies has no effect on the parent
- Cut boundaries in subspecies can not exceed the boundaries of the parent species.
- The *active* property is inherited also.

Most functions that can be applied to burst selections can be accessed by *right-clicking* the species list. Additionally, the species list contains a few buttons in the top right for quick access. These are, from left to right, shortcuts for adding a species, removing a species, renaming a species, exporting the selected species to *TauFit*, exporting the selected species to *PDAFit*, and enabling the multiselection mode to compare data sets (see *multiselection*).

Manual Cut

Manual Cut is available using the crop symbol or by pressing *ctrl+space* when the main plot is selected. This will turn the cursor into a crosshair that you can use to drag a rectangular selection on the current plot. The defined boundaries will then be applied to the current selected species.

Arbitrary Region Cut (AR)

Arbitrary Region Cut is available by clicking the free selection button or pressing *space*. Draw a shape in the main plot encircling the population you want to select and confirm the selection by *double-clicking* the plot. If instead you want to exclude a species from the selection, *right-click* the **Arbitrary Region Cut** button and select *Invert Arbitrary Region Cut*.

Arbitrary Region Cuts can not be modified after they have been defined. In the **Cut Table**, they will be abbreviated by *AR: *, followed by the parameter pair they have been defined on. The *min* and *max* fields have no effect for *AR* cuts. Like all cuts, if applied to a parent species, they will be automatically applied to the respective child species.

8.1.3 Storing cuts

If you have a set of cuts that you routinely apply to your data (i.e. to select donor-only molecules, or to remove bleaching artifacts…), you can store the cut state and apply it to newly loaded data. *Right-click* the **Cut Table**, select *Store in cut database* and specify a name for the cut state (e.g *Donly*). Keep the name simple since it will be converted into a valid MATLAB variable name, i.e. all spaces and special symbols will be removed.

After you have defined the cut, it will be shown in cut database popupmenu. To make use of the defined cut state, you can either apply it to the current species using the paint bucket, which will overwrite any existing cuts on the same parameters in the currently selected species, or add a new species with the stored cuts using the plus symbol.

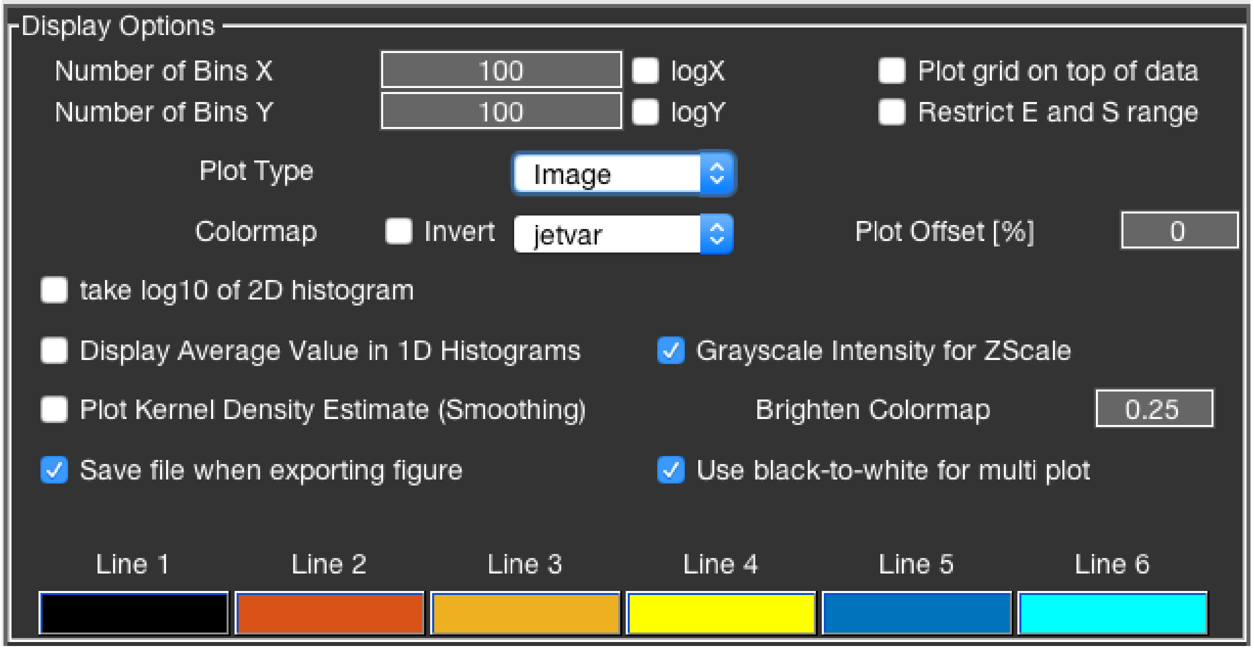
To remove a cut state from the database, *right-click* the popupmenu and select *Remove Cut from Database*. To show which cuts a certain stored cut state has defined, click *Displayed Selected Cut from Database* in the *right-click* menu. This will show a window with the cut information, and print it to the command line as well.

Note: *Arbitrary Region Cuts* are not stored in the cut database and simply ignored when saving a cut state.

8.1.4 Synchronizing cuts

When performing a measurement series, you may want to apply the same cuts to a number of files. When multiple files are loaded, you can synchronize cut states between the files by *right-clicking* the **Cut Table** and selection *Apply current cuts to all loaded files*. The cuts of the currently selected species will be applied to all other loaded files. If the file contains a species by the same name, the respective cuts will be overwritten/extended. If the file does not contain a species of the same name, a new species will be created for this file.

8.1.5 Customizing plots



Display Options

Number of Bins X 100 □ logX □ Plot grid on top of data

Number of Bins Y 100 □ logY □ Restrict E and S range

Plot Type Image

Colormap □ Invert jetvar Plot Offset [%] 0

□ take log10 of 2D histogram

□ Display Average Value in 1D Histograms ☑ Grayscale Intensity for ZScale

□ Plot Kernel Density Estimate (Smoothing) Brighten Colormap 0.25

☑ Save file when exporting figure ☑ Use black-to-white for multi plot

Fig. 3: The panel for selecting display options.

BurstBrowser's plotting functionality and data representation can be modified in a multitude of ways.

Bin numbers for the two-dimensional histograms can be changed independently for x and y dimensions. These settings globally affect the histograms in the "General" and "Lifetime" tabs. Additionally, the logarithm of the x- or y-parameter can be displayed by checking the logX or logY checkboxes.

The **plot type** for two-dimensional histograms can be chosen between *Image*, *Contour*, *Scatter* and *Hex*. See the section below section for more details on the plot types.

The **colormap** can be chosen from a range of standard MATLAB colormaps (jet, hot, bone, gray, parula, etc). Additionally, a variant of the jet colormap is available (jetvar), which starts from white before going through jet's color palette. All colormaps can be *inverted* by clicking the checkbox.

To plot the grid lines on top of the data, check the **Plot grid on top data** checkbox.

To make minor populations more visible, plotting of the **logarithm of the frequency information** of the two-dimensional histograms is supported. This option only applies to two-dimensional data, while not affecting the one-dimensional histograms.

By checking **Display Average Value in 1D Histograms** the average value and standard deviation of the respective parameter will be shown in the plots of the parameter histograms.

Enable **Save file when exporting figure** to automatically trigger a save dialog when closing an *export figure*.

The **Line** options allow to set custom colors for lines (i.e. static/dynamic FRET lines, Perrin plots, Gaussian fits). Click the colored box and select a color from dialog to change the line colors.

The other options are specific to other functions of **BurstBrowser** and are explained in the respective sessions.

Plot types

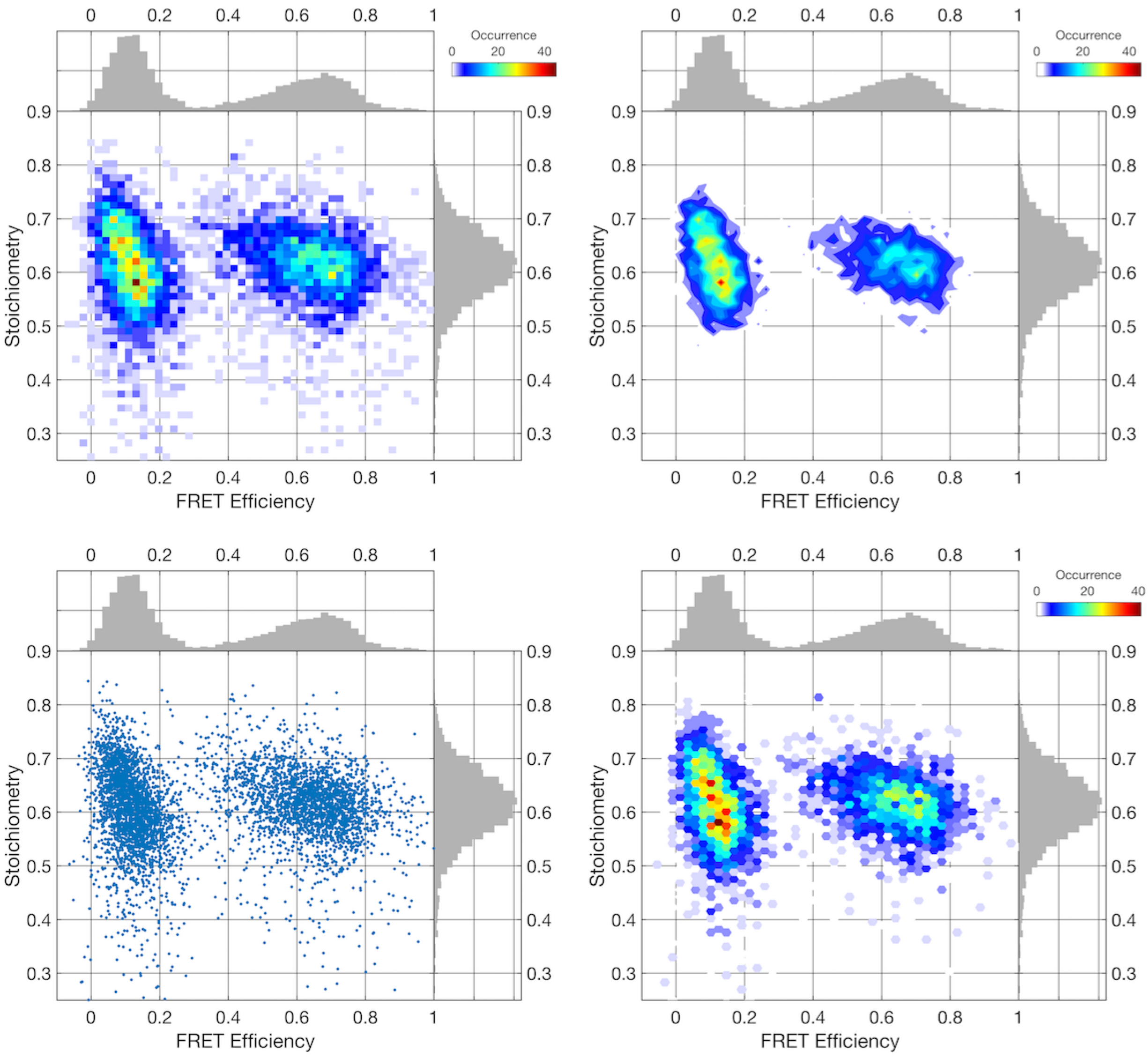


Fig. 4: The different plot types supported in BurstBrowser. From top left to bottom right: image, contour, scatter, and hex plot.

Image: Standard image plot. Can be used with *color by parameter*. The *Plot Offset [%]* parameter allows to hide all bins below the given percentage of the maximum value.

Contour: Contour plot that interpolates the binned data. Can further be customized by specifying the *number of contour levels* and the *contour offset*. Contour offset ignores all data below the given threshold, producing cleaner plots, but potentially hiding minor populations. For initial data analysis, it is thus recommended to use image plot, or to set the contour offset to zero. *Contour lines* can be disabled by unchecking the *Plot Contour Lines* checkbox.

Scatter: Scatter plot showing each data point. Can be customized by specifying the *marker color* and *marker size*. Can be used with *color by parameter*.

Hex: This plot uses hexagonal binning to represent the two-dimensional parameter distribution. Hexagonal binning is a more fitting pattern for round population and can thus represent the distribution better. The plotting is based on hexscatter.m freely available at the MATLAB file exchange.

Kernel density estimation (Smoothing)

Instead of the raw data one can plot smoothed data based on Gaussian kernel density estimation by checking *Plot Kernel Density Estimate (Smoothing)*. This can make it easier to detect populations if low amount of data is available. For kernel density estimation, an important parameter is the width of the Gaussian kernels. We apply the method described in [Botev2010] to determine an appropriate kernel bandwidth automatically from the data, using the script freely available at the MATLAB File Exchange. Kernel density estimation only works with *Image* and *Contour* plots. For *Scatter* and *Hex* plots the smoothing is only applied to the one-dimensional histograms.

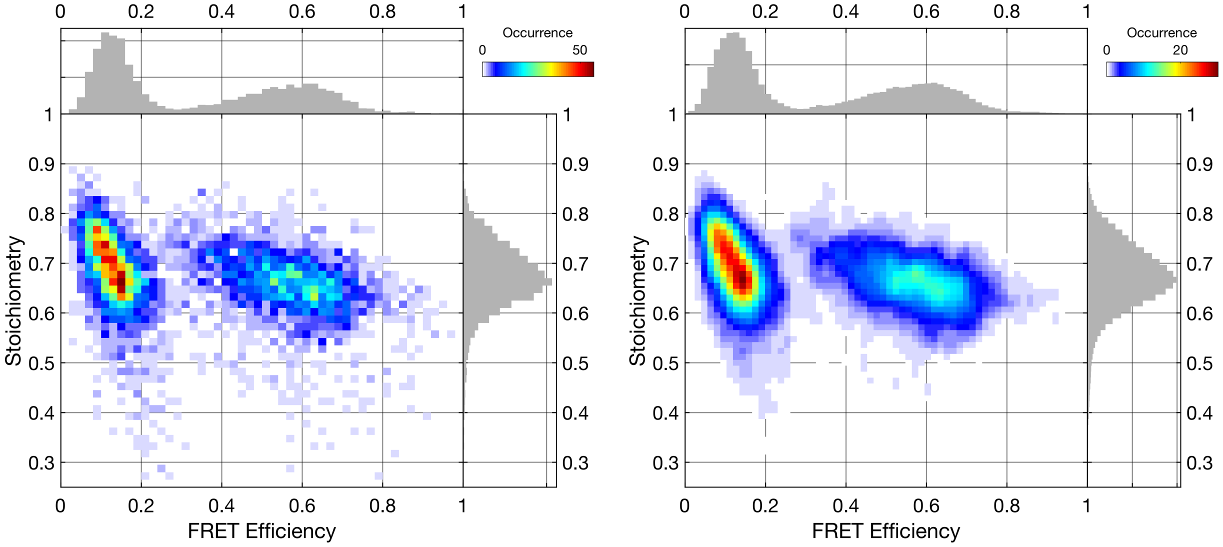
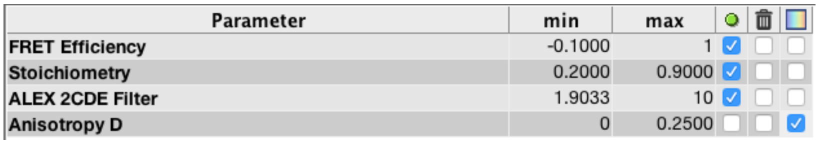


Fig. 5: Example of kernel density estimation (left: raw data, right: kernel density estimate)

Color plot by third parameter

Sometimes it can be helpful to investigate three parameters simultaneously. You can color-code a third parameter in the two-dimensional plot using the last column in the cut parameter table (*colorbar* icon).



Parameter	min	max	●	🗑	▨
FRET Efficiency	-0.1000	1	☑	□	□
Stoichiometry	0.2000	0.9000	☑	□	□
ALEX 2CDE Filter	1.9033	10	☑	□	□
Anisotropy D	0	0.2500	□	□	☑

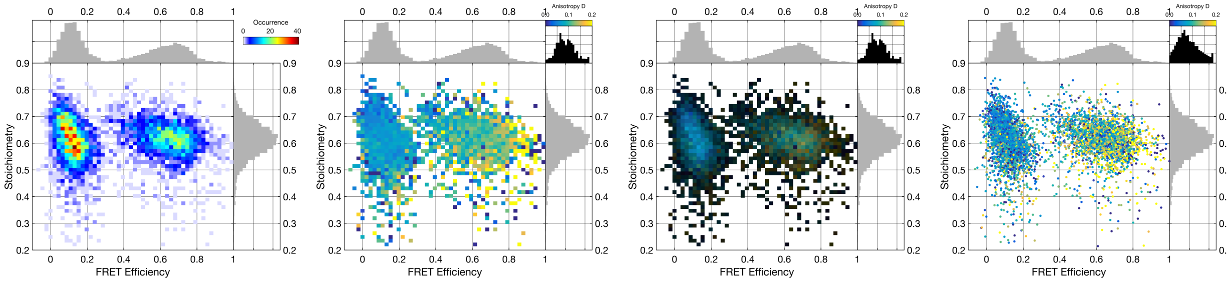


Fig. 6: Using a third parameter to color-code the E-S plot using the donor anisotropy. From left to right: normal intensity scaling, color-coded without and with intensity scaling, and using the scatter plot. The high FRET efficiency shows higher values for the donor anisotropy due to the shorter donor lifetime.

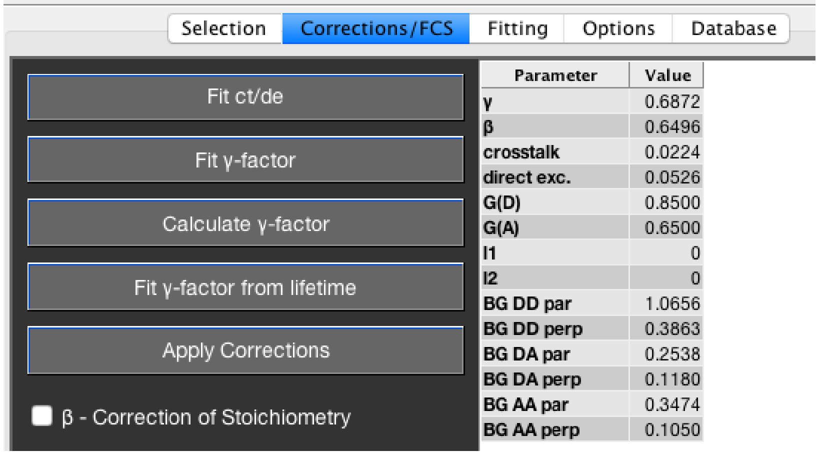
Color-coding is supported for *Image* and *Scatter* plots. The scaling is adjusted to the cut limits of the color-coding parameter. Use the parameter histogram in the top right to adjust the limits. When using an image plot, the parameter histogram for the color-coding parameter in the top right will show the distribution of the pixel-wise average, and thus be different from the burst-wise histogram of the same parameter. For scatter plots, the true burst-wise distribution is shown, since no averaging is needed for color-coding. Thus, when using the image plot, it is advised to set the color-coding parameter cut state to inactive to avoid removing data points when adjusting the cut limits.

You can combine color-coding and intensity information by checking the *Intensity scaling for color coding* checkbox. Additionally, you can brighten the intensity scaling using the *Brighten Colormap* option. Color-coding is not compatible with multiselection.

8.2 Correcting data

To obtain accurate FRET efficiencies, a number of *correction factors* is required. Crosstalk, direct excitation and γ-factor can be determined from the measurement directly if PIE is used.

The correction factors are all entered into the **Corrections Table** found in the *Corrections/FCS* tab.



Parameter	Value
γ	0.6872
β	0.6496
crosstalk	0.0224
direct exc.	0.0526
G(D)	0.8500
G(A)	0.6500
l1	0
l2	0
BG DD par	1.0656
BG DD perp	0.3863
BG DA par	0.2538
BG DA perp	0.1180
BG AA par	0.3474
BG AA perp	0.1050

Fig. 7: The corrections panel.

On first load, the last used corrections are automatically filled into the corrections table. Background count rates are those determined from the specified scatter measurement. If you want corrections to automatically be applied upon load, check the *Automatically apply default/stored corrections when loading a file* checkbox under *Options->Data Processing Options*.

8.2.1 Changing parameters

Any parameter can be edited in the **Corrections Table**, but the changes will only be applied after the *Apply Corrections* button has been clicked. The *Apply Corrections* button will turn red to indicate that changes have been performed but not confirmed yet.

The β-factor accounts for different excitation efficiencies of donor and acceptor fluorophore, which cause the stoichiometry value of double labeled molecules to deviate from the theoretical value of 0.5. For a correctly chosen β-factor, the stoichiometry is exactly 0.5 for a double-labeled species. The correction of the stoichiometry value by the β-factor can be toggled using the respective checkbox.

Note: Correction of the stoichiometry by usage of the β-factor is advised if different interaction stoichiometries are investigated. β-corrected stoichiometries allow to infer interaction stoichiometries directly, i.e. $S = 0.5$ corresponds to a 1:1 stoichiometry, while $S = 0.25$ or $S = 0.75$ correspond to a stoichiometry of 1:3 or 3:1 of donor to acceptor fluorophores, respectively.

8.2.2 Determination of correction factors

Crosstalk and Direct Excitation

To automatically determine the *crosstalk* and *direct excitation* correction factors from donor- and acceptor only populations, click the *Fit ct/de* button. Donor- and acceptor-only populations will be selected based on the stoichiometry (raw) thresholds specified under *Options->Data Processing Options*. To isolate the acceptor only population, additionally a minimum threshold for the proximity ratio (i.e. the uncorrected FRET efficiency) can be set. To determine proper values for the thresholds, inspect of *Stoichiometry (raw)* vs. *Proximity Ratio*.

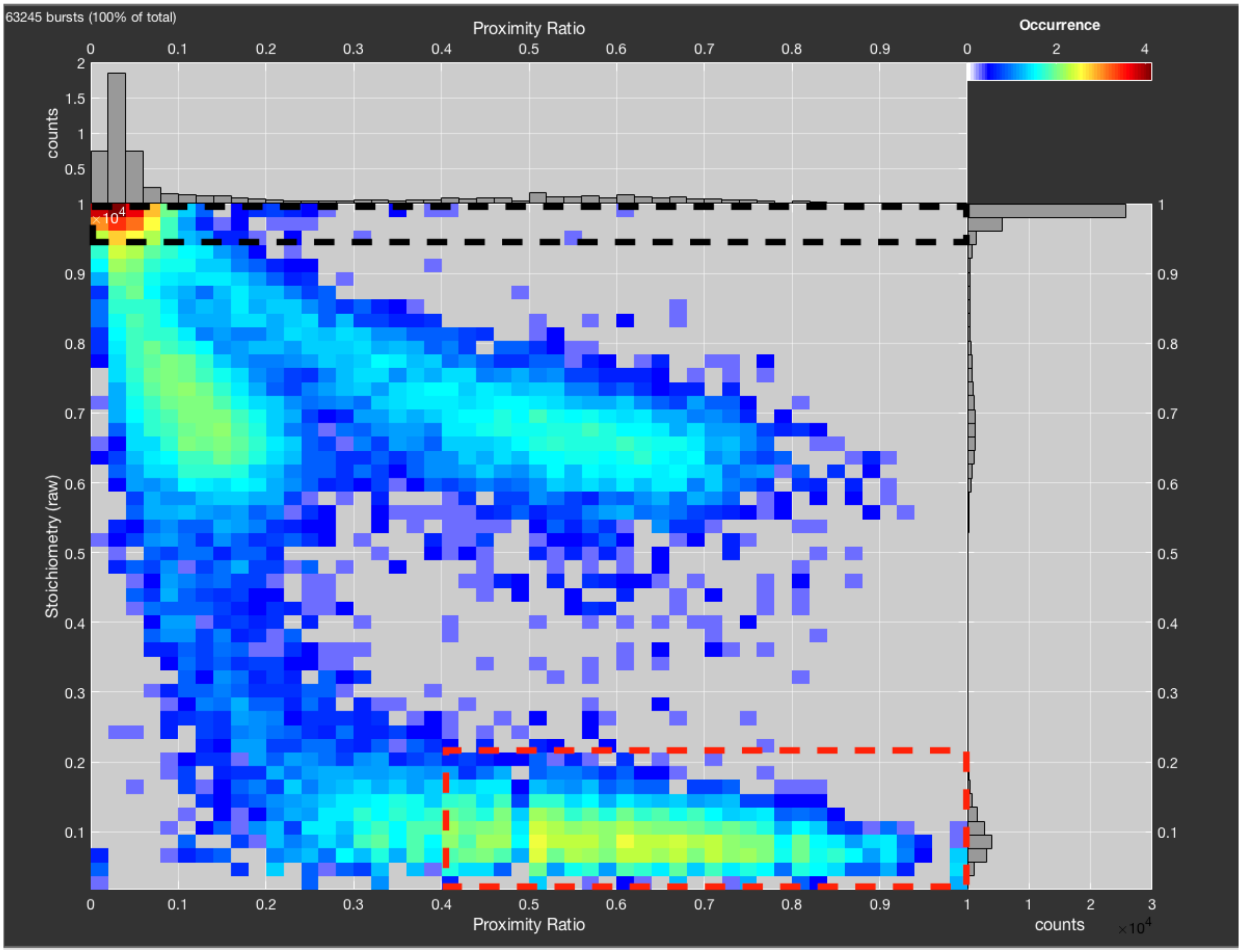


Fig. 8: Logarithmic plot of raw stoichiometry vs. proximity ratio to choose thresholds for donor-only (black) and acceptor-only (red) populations. Donor-only is selected by requiring S > 0.95. Acceptor-only is selected by S < 0.22 and proximity ratio > 0.4. Without the proximity ratio the acceptor-only population would be contaminated by molecules from the bridge between acceptor only and low FRET efficiency population.

Histograms of the proximity ratio of the donor-only population and the raw stoichiometry of the acceptor-only populations will be fit with a single Gaussian function to determine the center value. The result is plotted in the **Corrections** tab, which is automatically given priority. If a single Gaussian is not sufficient to fit the data (adjusted $R^2 < 0.9$), two Gaussian functions are used to fit the data and the center value of the main contribution is used.

Crosstalk and direct excitation are determined from the obtained values E_{PR}^{D-only} and S_{raw}^{A-only} by:

$$ct = \frac{E_{PR}^{D-only}}{1 - E_{PR}^{D-only}}$$

$$de = \frac{S_{raw}^{A-only}}{1 - S_{raw}^{A-only}}$$

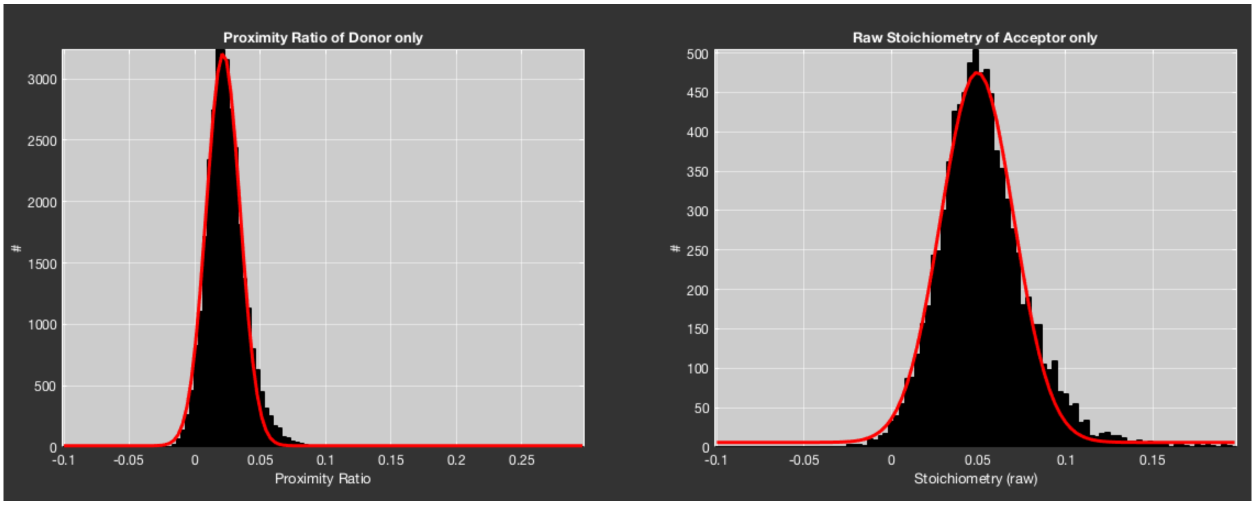


Fig. 9: Example for the determination of crosstalk and direct excitation correction factors.

When the parameters for crosstalk or direct excitation are changed following automatic determination, the fits in the plots in the **Corrections** tab will be adjusted accordingly. Use this to judge if your chosen values are in agreement with the data (i.e. if values have been determined from separate dye measurements).

γ-factor

The γ-factor corrects for different detection efficiencies and quantum yields of donor and acceptor fluorophore. It can be determined from the data in two ways, either using the dependency of FRET efficiency and stoichiometry, or by using the fluorescence lifetime information of the donor fluorophore.

The corrected stoichiometry value should be independent of the observed FRET efficiency, i.e. if the γ-factor is chosen correctly, it the stoichiometry is unaffected by the distribution of the fluorescence signal on the two detection channels. This fact can be used to determine the γ-factor from a measurement of multiple FRET species. For $\gamma = 1$, a linear relation is obtained between the inverse of the stoichiometry and the FRET efficiency [Lee2005]. The fit parameters of the linear fit $\frac{1}{S} = \Omega + \Sigma \cdot E$ are related to the γ-factor and the β-factor (accounting for different excitation efficiencies, see *correction factors*):

$$\gamma = \frac{\Omega - 1}{\Omega + \Sigma - 1}$$

$$\beta = \Omega + \Sigma - 1$$

The determination of the γ-factor by this method is implemented in two ways. **Fit γ factor** uses a burst-wise plot of $1/S$ vs. E for $\gamma = 1$ to perform a linear fit using the *fit* function of MATLAB. To reduce the effect of extreme outliers that occur at small stoichiometry values when taking the inverse, robust fitting by the least absolute residual (LAR) method is applied. The fit result is shown in the bottom left plot in the *Corrections* tab.

Alternatively, the γ-factor can be calculated directly without fitting if the center points of two different FRET species in the E-S plot are determined. *Right-click* the **Fit γ factor** button and select *Determine γ factor manually* to turn the mouse cursor into a crosshair that you can use to select the centers of two different FRET species sequentially in the E-S plot shown in the bottom left plot of the *Corrections* tab.

The most robust way to determine the γ-factor is to determine the E-S values for $\gamma = 1$ for different populations using the *Gaussian fitting module*. Once determined, you can calculate the γ-factor by clicking the **Calculate γ factor** button and inputting the values into the table. Optionally, one can specify the uncertainty for E and S (σE and σS), which will be used to estimate the error in the determined γ- and β-factors. Note that the uncertainty in E and S is not given by the standard deviation as obtained from the fit. A more appropriate estimate for the uncertainty would be the standard error of the mean given by $SEM = \sigma/\sqrt{N}$, where N is the number of data points, i.e. bursts, used for the fit.

If the fluorescence lifetime information of the donor is available, you can determine γ even if only a single FRET population is available, as long as the measured system is static, i.e. shows no FRET dynamics (e.g. a DNA molecule). Clicking **Fit γ factor from lifetime** will minimize the deviation of the FRET species from the theoretical static FRET line defined by the donor-only lifetime, the Förster radius and the observed linker length. The fit routine uses robust fitting by the bisquare method. The result is displayed in the bottom left plot of the *Corrections* tab. For more information on the dependence between the FRET efficiency and the fluorescence lifetime of the donor, see the respective section on the *static FRET line*.

Note: The γ-factor can also be determined by inspecting the E-τ_D plot and manually varying γ until the population falls onto the static FRET line. This is advised if the automatic fitting does not yield satisfying results.

In all cases, the currently selected bursts are used for the automatic determination. Ensure that only double labeled molecules are selected. If no single measurement with multiple FRET species is available, you can combine separate measurements of different FRET species using the *multiplot* functionality to plot them simultaneously. The automatic determination will use the currently selected bursts, and combine the selection over multiple files.

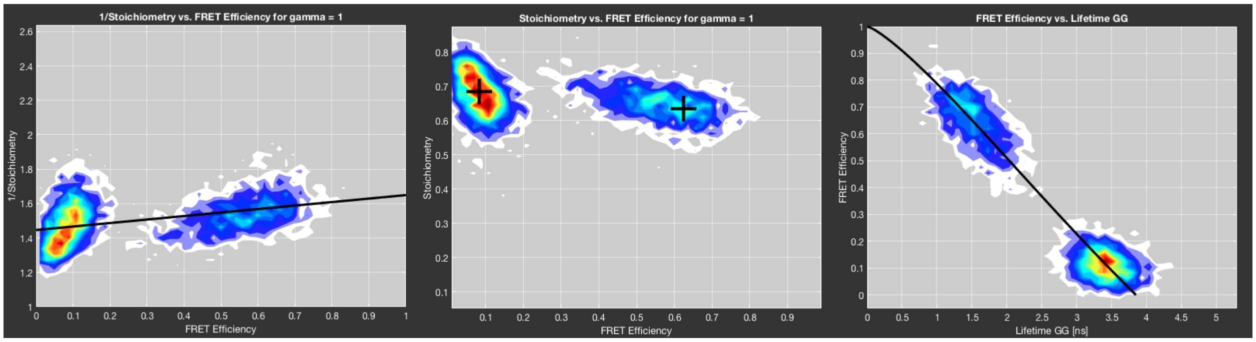


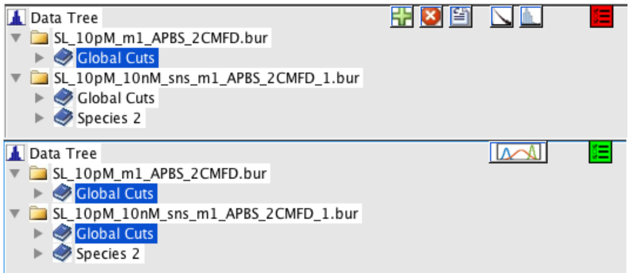
Fig. 10: Determining the γ-factor. (Left) Fit of 1/S vs. E. (Middle) Manual selection of the center points of the population. (Right) Using the fluorescence lifetime of the donor fluorophore.

8.2.3 Synchronizing correction factors

To synchronize correction factors between all loaded files, *right-click* the *Apply Corrections* button and select *Replace corrections of all files with current one*. This will overwrite the correction parameters of all other files with the current one.

8.3 Comparing datasets

8.3.1 Multiselection



Data Tree
- SL_10pM_m1_APBS_2CMFD.bur
  - Global Cuts
- SL_10pM_10nM_sns_m1_APBS_2CMFD_1.bur
  - Global Cuts
  - Species 2

Data Tree
- SL_10pM_m1_APBS_2CMFD.bur
  - Global Cuts
- SL_10pM_10nM_sns_m1_APBS_2CMFD_1.bur
  - Global Cuts
  - Species 2

Fig. 11: The MultiSelection button in the top right of the Species List toggles the multiselection mode. Top: multiselection disabled, bottom: multiselection enabled. Enabling the multiselection mode display the MultiPlot button, while hiding the buttons associated with single species.

Multiselection mode is activated by clicking the *MultiSelection* button in the *Species List* (see *Species List*), which will allow multiple species in the *Species List* to be selected simultaneously. The button turns red when *MultiSelection* is disabled, and green when it is enabled. This will disable all options associated with the *Species List* that are related to single files. Multiselection works between different files.

Selecting multiple species simultaneously will superimpose all selected datasets. Multiselection works with all functions to determine *correction factors*. While the two-dimensional plot will only show the superposition of all selected species, the marginal distributions will show histograms for the individual species. When using the *Scatter* plot, species will additionally be color-coded in the two-dimensional plot.

By default, the selected species are area-normalized. This is ensures that all species are visible, even if there are large differences in the number of bursts per species. The histograms can alternatively be normalized to their maximum value by *right-clicking* the *MultiSelection* button and selecting *Normalize to…* -> *maximum*. You can disable the normalization by *right-clicking* the *Multiselection* checkbox and unchecking *Normalize populations*. The display of the sum of all selected populations can be toggled in the same *right-click* menu by deselecting the *Display sum of all populations* option.

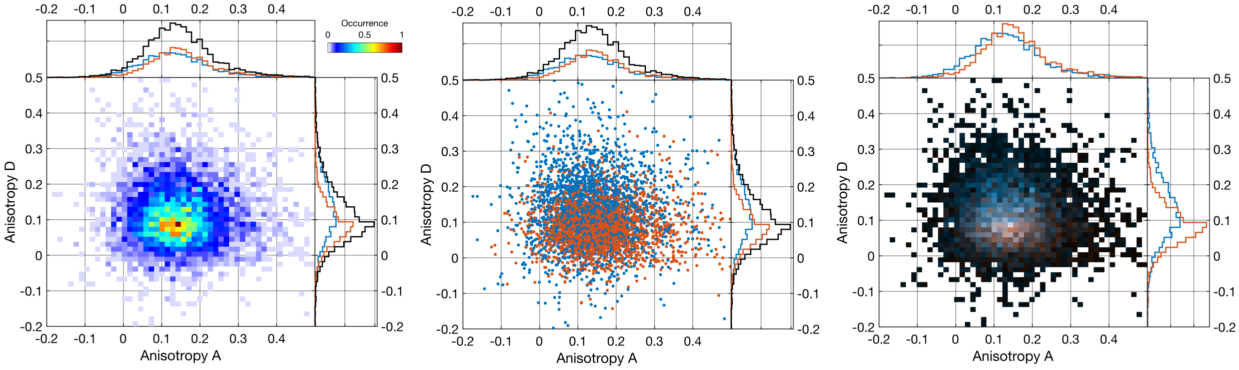


Fig. 12: Multiselection using *Image* plot will superimpose the populations in the two-dimensional plot, but show the individual populations in the marginal distributions. The *Scatter* plot shows the different populations using different colors. The *Multiplot* functionality color-coded the populations.

The **Multiplot** button allows for up to three species to be plotted simultaneously using a separate color for each species (blue, red and green in that order). It appears in the top right of the *Species List* when the *MultiSelection* mode is activated. If more than three species are selected, the first three will be used. Two coloring modes are available that can be changed under *Options->Display Options*. The default mode uses RGB channels to display the different species. Occurrence is encoded by color intensity for every color channel, starting from white. High intensity regions of overlapping species will be black in this mode. Alternatively to RGB, a “black-to-white” mode is implemented, which starts at black color for low intensity going to white at high intensity.

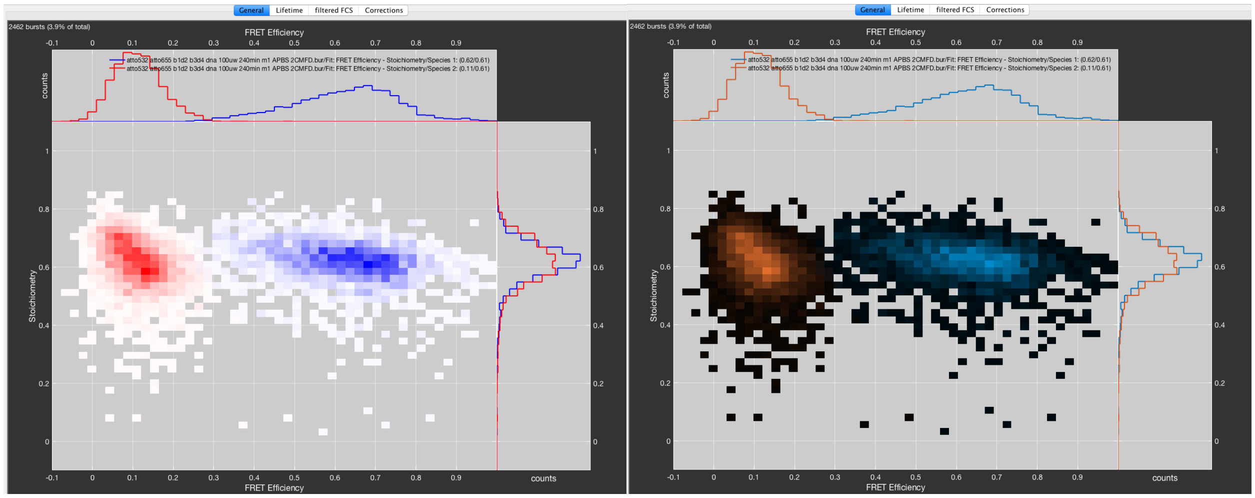

General
Lifetime
filtered FCS
Corrections
2462 bursts (3.9% of total)
FRET Efficiency
-0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
atto532 atto655 b1d2 b3d4 dna 100uw 240min m1 APBS 2CMFD.bur/Fit: FRET Efficiency - Stoichiometry/Species 1: (0.62/0.61)
atto532 atto655 b1d2 b3d4 dna 100uw 240min m1 APBS 2CMFD.bur/Fit: FRET Efficiency - Stoichiometry/Species 2: (0.11/0.61)
counts
Stoichiometry
1
0.8
0.6
0.4
0.2
0
counts
FRET Efficiency

Fig. 13: Multiplot of two species using white-to-black mode (left) and black-to-white mode (right).

Hint: If more than three species need to be visualized simultaneously, it is best to use a *Scatter* plot together with multiselection.

8.3.2 Comparing single parameters

The most common case when analyzing multiple data sets is to compare the FRET efficiency histograms of separate measurements. This is available by clicking through the *Compare* menu in the toolbar. Select *Compare FRET histograms of loaded files* to create a single plot with all FRET efficiency histograms, using the last selected species of the respective files. If another parameter is to be compared, select *Compare current parameter of loaded files* instead. This will use the currently selected parameter for the *x*-axis, i.e. in the left parameter selection list.

To compare parameters of different species from the same measurement (or between measurements), enable the *Multiselect* checkbox and select the species to be compared. *Compare -> Compare current parameter of selected species* will again use the currently selected *x*-axis parameter.

When comparing FRET efficiency histograms, e.g. from a time series, one can also display a waterfall plot by enabling the *Make waterfall plot when comparing FRET histograms* in the *Options -> Data processing* tab.

8.4 Using the lifetime information

The fluorescence lifetime information is an important parameter in burst analysis. It allows to identify photophysical artifacts like quenching, and, together with the anisotropy information, the rotational freedom of the fluorophore can be addressed. Furthermore, it serves as a parameter to identify conformational dynamics on the burst timescale.

BurstBrowser gives an overview over the most important lifetime-related plots in PIE-MFD in the *Lifetime* tab. These plots should be carefully inspected for all PIE-MFD measurements.

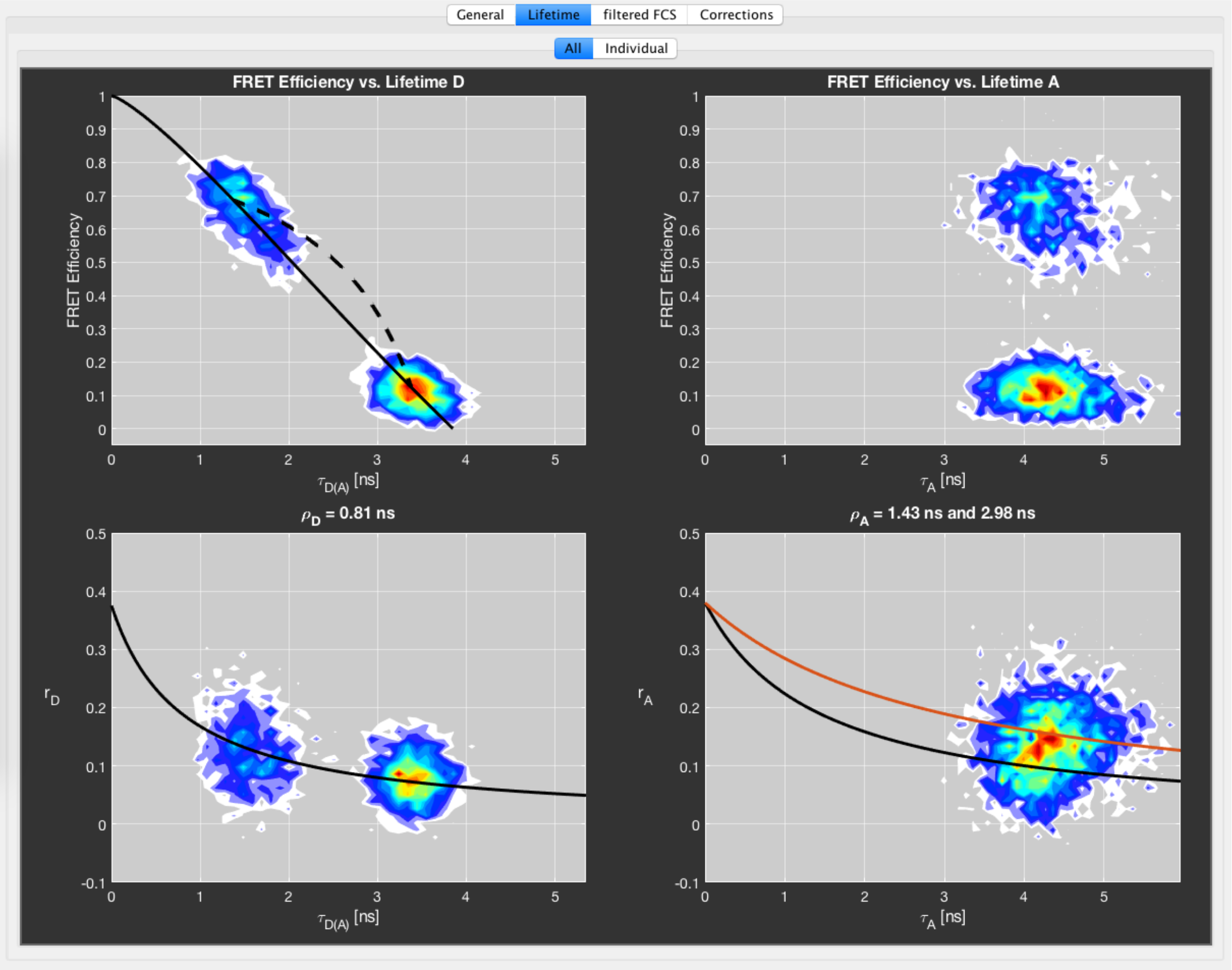
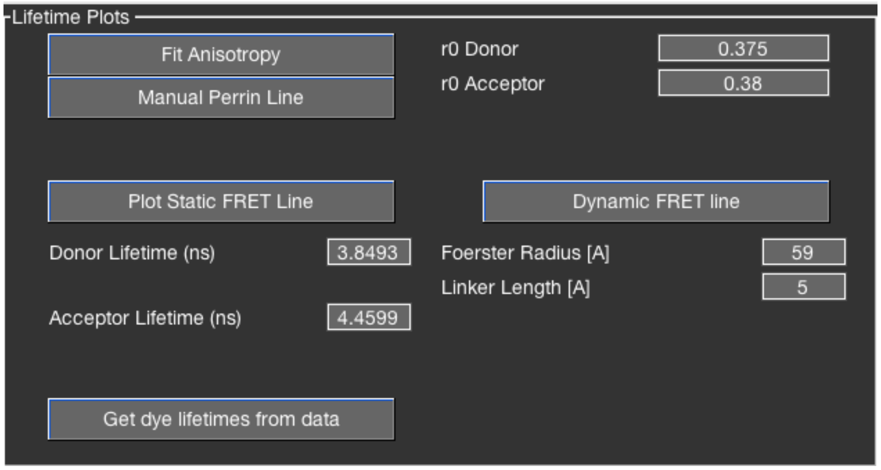


Fig. 14: The *Lifetime* tab showing all important lifetime-related plots.



r0 Donor 0.375
r0 Acceptor 0.38
Donor Lifetime (ns) 3.8493
Acceptor Lifetime (ns) 4.4599
Foerster Radius [A] 59
Linker Length [A] 5

Fig. 15: The settings for fits in the *Lifetime* tab, found under the *Fitting* tab.

In addition to the overview given in the *Lifetime -> All* tab, the *Individual* tab allows to inspect the lifetime-related plots individually using a larger plot with additional marginal distributions. Use the list in the top right to change the current plot. All fits can also be performed in the *Individual* plot.

8.4.1 E vs. $\tau_{D(A)}$

The top-left plot shows the FRET efficiency plotted versus the fluorescence lifetime of the donor (in presence of the acceptor, $\tau_{D(A)}$). In theory, the dependence between FRET efficiency is linear and given by the so called "static FRET line":

$$E = 1 - \frac{\tau_{D(A)}}{\tau_{D,0}}$$

where $\tau_{D,0}$ is the lifetime of the donor in the absence of the acceptor. Practically, the plot deviates from the theoretical line because the dyes are attached to linkers, leading to fast fluctuations (~100 ns) in the donor-acceptor distance even for static samples that are averaged on the burst timescale (~ms). Since the lifetime fit is performed only on the donor fluorescence decay, low FRET efficiency states contribute more photons to the decay, and the determined lifetime is thus biased towards long lifetimes. This effect leads to a slight curving of the theoretical static FRET line, which depends on the Förster radius and the apparent length of the linker, i.e. the magnitude of the distance fluctuations. The apparent linker length can be determined from a TCSPC analysis of a static FRET sample by fitting a *distance distribution*, and is usually on the order of 5-6 Angstrom.

Equivalently, conformational dynamics during bursts will lead to mixture of states. While the FRET efficiency is averaged based on the time the molecule spent in the states, the fluorescence lifetime is again biased by the brightness of the species and thus biased towards the low FRET efficiency, long fluorescence lifetime state. In the absence of linker dynamics, the dynamic FRET lines between two states (1 and 2) with different FRET efficiencies is given by:

$$E = 1 - \frac{\tau_1 \tau_2}{\tau_{D,0}\left(\tau_1 + \tau_2 - \langle\tau\rangle_f\right)}$$

Here, $\langle\tau\rangle_f$ is the intensity-weighted lifetime as determined by the burst-wise fit. Note that there is no analytical expression for static or dynamic FRET lines when linker dynamics are to be considered.

To correctly account for the linker dynamics, the static and dynamic FRET lines in **BurstBrowser** are simulated as described in the supplementary information to [Kalinin2010], taking the linker dynamics into account. However, in contrast to what is described in [Kalinin2010], no approximation by a third-order polynomial is performed in **BurstBrowser**; instead, the calculated lines are used directly without the intermediate step of approximation because the simulation is fast.

Plotting the static FRET line

The static FRET line is a function of the donor-only lifetime, the Förster radius and the apparent linker length. It can be plotted by clicking the *Plot static FRET line* button and is updated when any of the related parameters are modified.

Using the dynamic FRET line

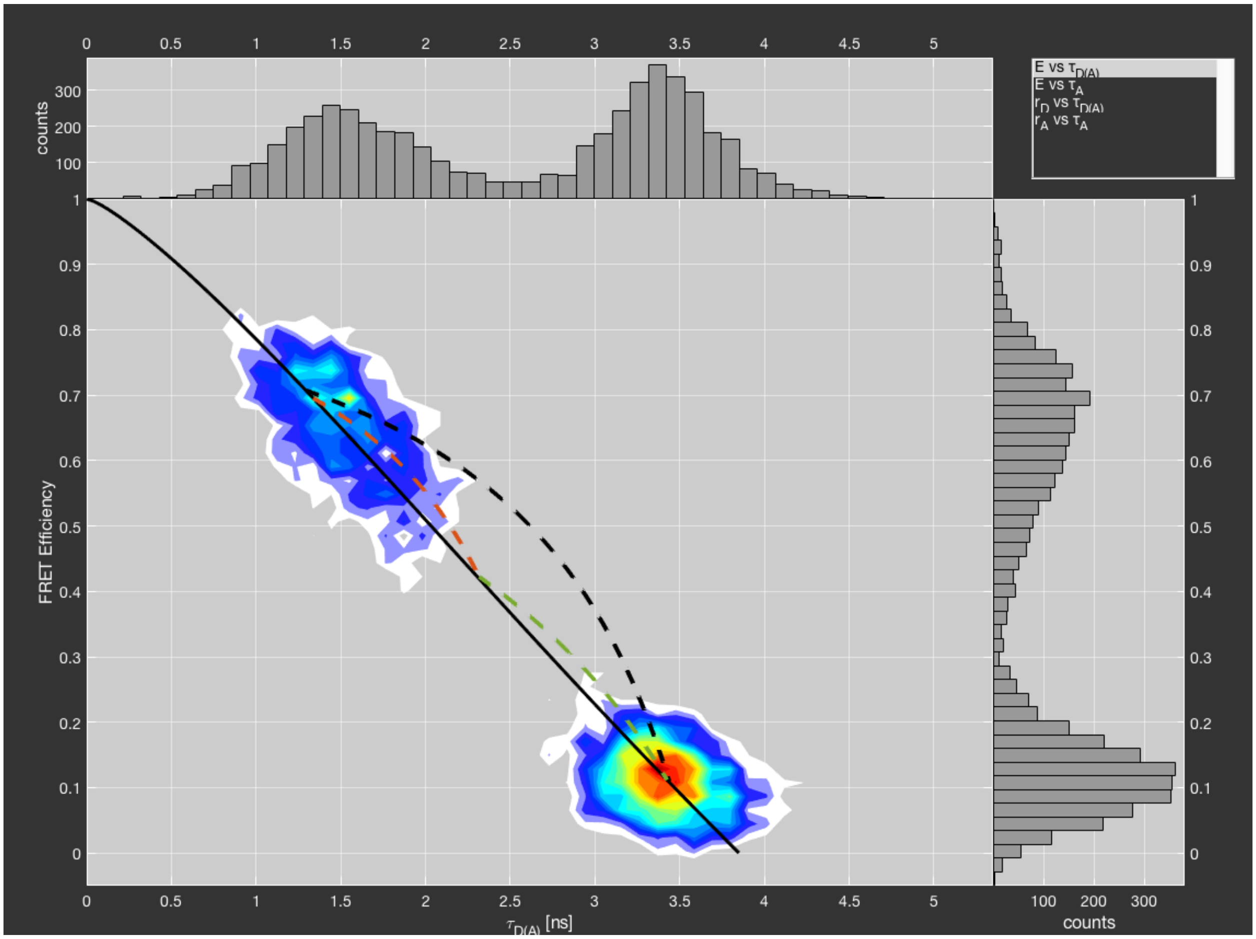


Fig. 16: An example of static and dynamic FRET lines. Dynamic FRET lines are plotted between the two states (black) and additionally for a hypothetical intermediate state (red and green).

Click the *Dynamic FRET line* to define a dynamic FRET line. This will turn the cursor into a crosshair, allowing you to define the two states belonging to the dynamic FRET line by clicking into the respective plot. You can plot up to three dynamic FRET lines together. Add more dynamic FRET lines by *right-clicking* the plot instead of *left-clicking*.

Alternatively, you can *right-click* the button and select *Define States* to manually define the two states by their respective lifetime. Choose which line you want to define, up to three lines maximum. Remove individual lines by *right-click* the *Dynamic FRET line* button and selecting *Remove Plot* to specify the number of the line you want to remove. Line colors correspond to the settings in the *Options* tab. Dynamic FRET lines will not updated if related parameters are changed.

Hint: Use *sub-ensemble TCSPC* and a biexponential lifetime fit to determine the fluorescence lifetimes of the two states if the dynamics are fast.

Getting donor-only and acceptor-only lifetimes

Donor- and acceptor-only lifetimes can be extracted from the donor- and acceptor-only subpopulations of the experiment. This gives an intrinsic reference for the static FRET line and removes the necessity to determine the donor-only lifetime from a reference sample.

The donor- and acceptor-only lifetimes can be determined in two principal ways: from the burst-wise lifetime distributions, or from a sub-ensemble TCSPC analysis.

The former is automated in BurstBrowser. Simply click the *Get dye lifetimes from data* button in the *Fitting* tab (subsection *Lifetime Plots*). This will automatically select donor- and acceptor-only populations and determine the mean burst-wise lifetime by Gaussian fitting. The fit can be inspected in the *Corrections* tab in the top two plots (the same plots previously used for inspecting the quality of the *crosstalk and direct excitation fits*). The values for donor- and acceptor-only lifetimes will be automatically updated.

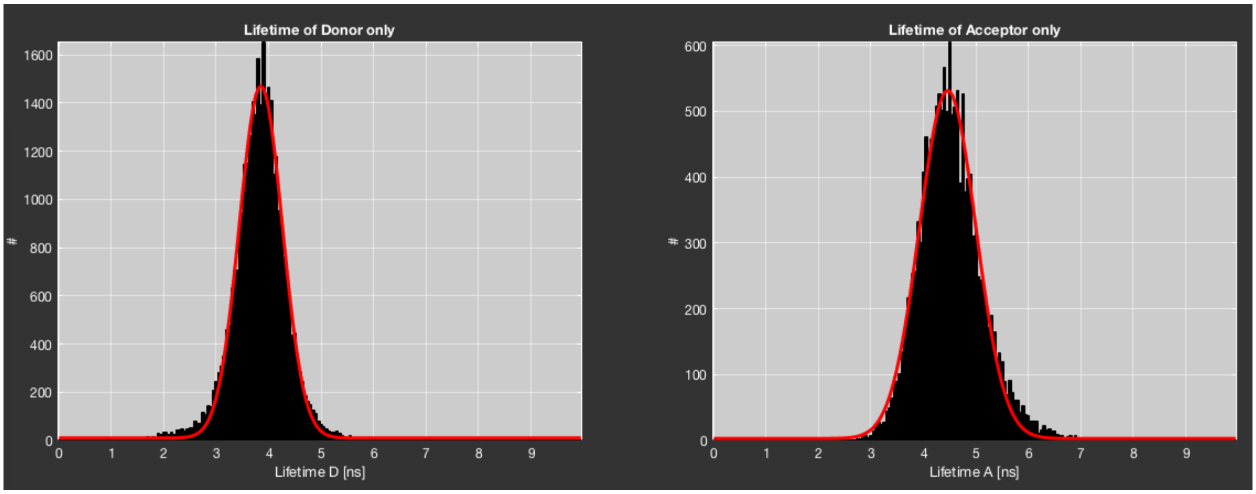


Fig. 17: Example fits of donor- and acceptor-only lifetime distributions.

Sub-ensemble TCSPC analysis can be manually performed on selected donor- and acceptor-only populations as described in the *corresponding section*. Determined lifetimes can be manually input into the edit boxes.

8.4.2 Anisotropy vs lifetime

The measured steady-state anisotropy depends on the fluorescence lifetime. Since the anisotropy itself decays, a shorter fluorescence lifetime will result in a higher averaged anisotropy than a long fluorescent lifetime at equal rotational correlation time. The dependance between anisotropy and lifetime is given by the Perrin equation, assuming single exponential lifetime and anisotropy decays:

$$r\left(\tau\right) = \frac{r_0}{1 + \frac{\tau}{\rho}}$$

Here, r_0 is the fundamental anisotropy of the fluorophore and ρ is the rotational correlation time.

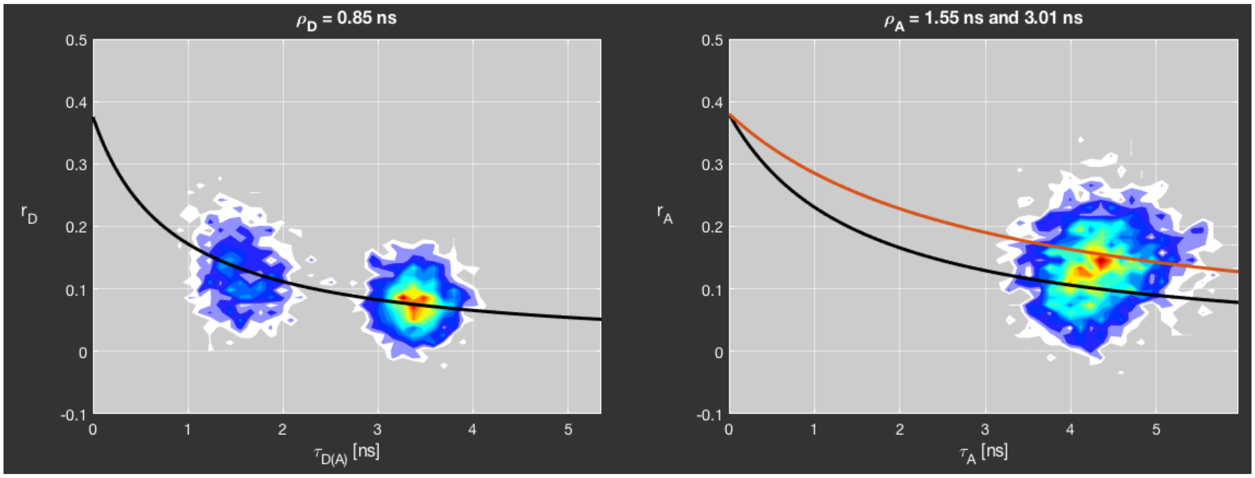


Fig. 18: An example of Perrin plots.

You can add Perrin lines to the anisotropy plots to distinguish species based on their rotational correlation time. Clicking the **Fit Anisotropy** button to fit Perrin lines to the $r - \tau$ distribution. Determined rotational correlation times will be shown above the plots. You can manually add Perrin lines by clicking the **Manual Perrin line** button and selecting a population in the plot. Add up to two more Perrin lines by repeating the previous step but *right-clicking* the plot. Perrin lines will use the specified r_0 values for the respective colors. Reset any present Perrin fits by either clicking the **Fit Anisotropy** button or defining a new primary Perrin line using the **Manual Perrin line** button.

8.4.3 Lifetime/anisotropy fitting by sub-ensemble TCSPC

You can perform sub-ensemble (or burst-selective) TCSPC analysis on a species by selecting it in the *species list* in the bottom right of the **Selection** tab, *right-clicking* and selecting **Send selected species to TauFit**. Alternatively, you can click the small fluorescence decay symbol at the top of the **species list**. This is only enabled if **Multiselect** is disabled, since you can only send single species to **TauFit**. This will open an instance of **TauFit** with the cumulative data of all selected bursts. You can use the whole toolset of **TauFit** as described *in the TauFit manual*.

Instead of selecting *PIE channels*, you will be able to select *FRET channels*, i.e. *DD*, *AA*, and *AD* for two-color FRET. **TauFit** will use the instrument response function and scatter/background pattern that is stored with the burst measurement. The FRET-sensitized acceptor decay (*DA*) is a convolution of FRET-quenched donor decay and acceptor decay, for which no model is supplied in **TauFit**. However, the residual anisotropy of donor, acceptor and FRET-sensitized acceptor emission can be used to gain information about the distribution of the orientational factor κ^2, as described in the *respective section of the TauFit manual*.

8.5 Fitting data using Gaussian distributions

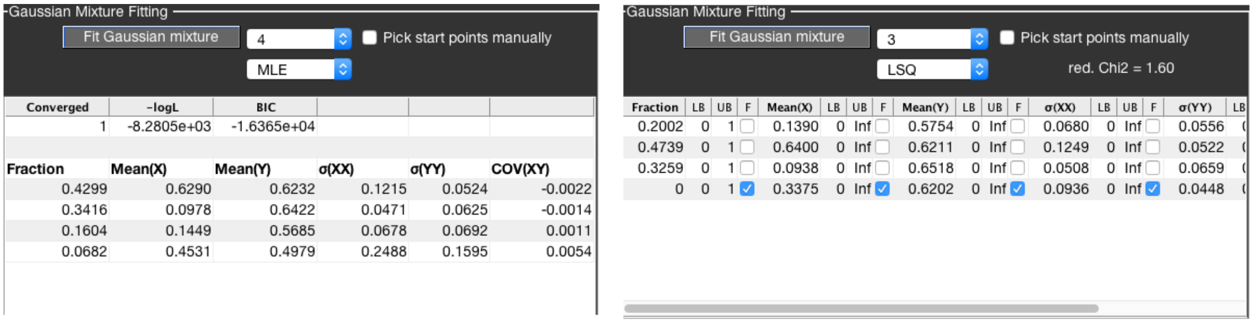
BurstBrowser allows to fit any combination of parameters in the **General** plot using a Gaussian mixture model with up to 4 components, using either a maximum likelihood estimator (MLE) or by minimizing the squared error (LSQ). Select the fit method from the bottom dropdown menu and the number of Gaussian distributions to be used from the top dropdown menu (currently, up to 4 species are available). Checking the **Pick start points manually** checkbox will allow you to manually specify start points for the fit algorithm by clicking in the plot.

The fit model M is given by a superposition of N bivariate normal distributions with amplitude A, mean value vector

μ and covariance matrix Σ:

$$M(x,y)=\sum_{i=1}^{N}A_i\cdot(2\pi)^{-1}\left|\Sigma_i\right|^{-1/2}\exp\left[-\frac{1}{2}\left(x-\mu_i\right)'\Sigma_i^{-1}\left(x-\mu_i\right)\right]$$

The parameters of the fit model are the fractional contributions of each species, the center points in *X* and *Y* dimensions *Mean(X)* and *Mean(Y)*, as well as the corresponding distributions widths $\sigma(XX)$ and $\sigma(YY)$ and covariance between the parameters *COV(XY)*.



Gaussian Mixture Fitting

Fit Gaussian mixture 4 □ Pick start points manually

MLE

Converged	-logL	BIC			
1	-8.2805e+03	-1.6365e+04			

Fraction	Mean(X)	Mean(Y)	σ(XX)	σ(YY)	COV(XY)
0.4299	0.6290	0.6232	0.1215	0.0524	-0.0022
0.3416	0.0978	0.6422	0.0471	0.0625	-0.0014
0.1604	0.1449	0.5685	0.0678	0.0692	0.0011
0.0682	0.4531	0.4979	0.2488	0.1595	0.0054

Gaussian Mixture Fitting

Fit Gaussian mixture 3 □ Pick start points manually

LSQ red. Chi2 = 1.60

Fraction	LB	UB	F	Mean(X)	LB	UB	F	Mean(Y)	LB	UB	F	σ(XX)	LB	UB	F	σ(YY)	LB
0.2002	0	1	□	0.1390	0	Inf	□	0.5754	0	Inf	□	0.0680	0	Inf	□	0.0556	
0.4739	0	1	□	0.6400	0	Inf	□	0.6211	0	Inf	□	0.1249	0	Inf	□	0.0522	
0.3259	0	1	□	0.0938	0	Inf	□	0.6518	0	Inf	□	0.0508	0	Inf	□	0.0659	
0	0	1	☑	0.3375	0	Inf	☑	0.6202	0	Inf	☑	0.0936	0	Inf	☑	0.0448	

Fig. 19: Gaussian mixture fitting user interface for MLE (left) and LSQ (right).

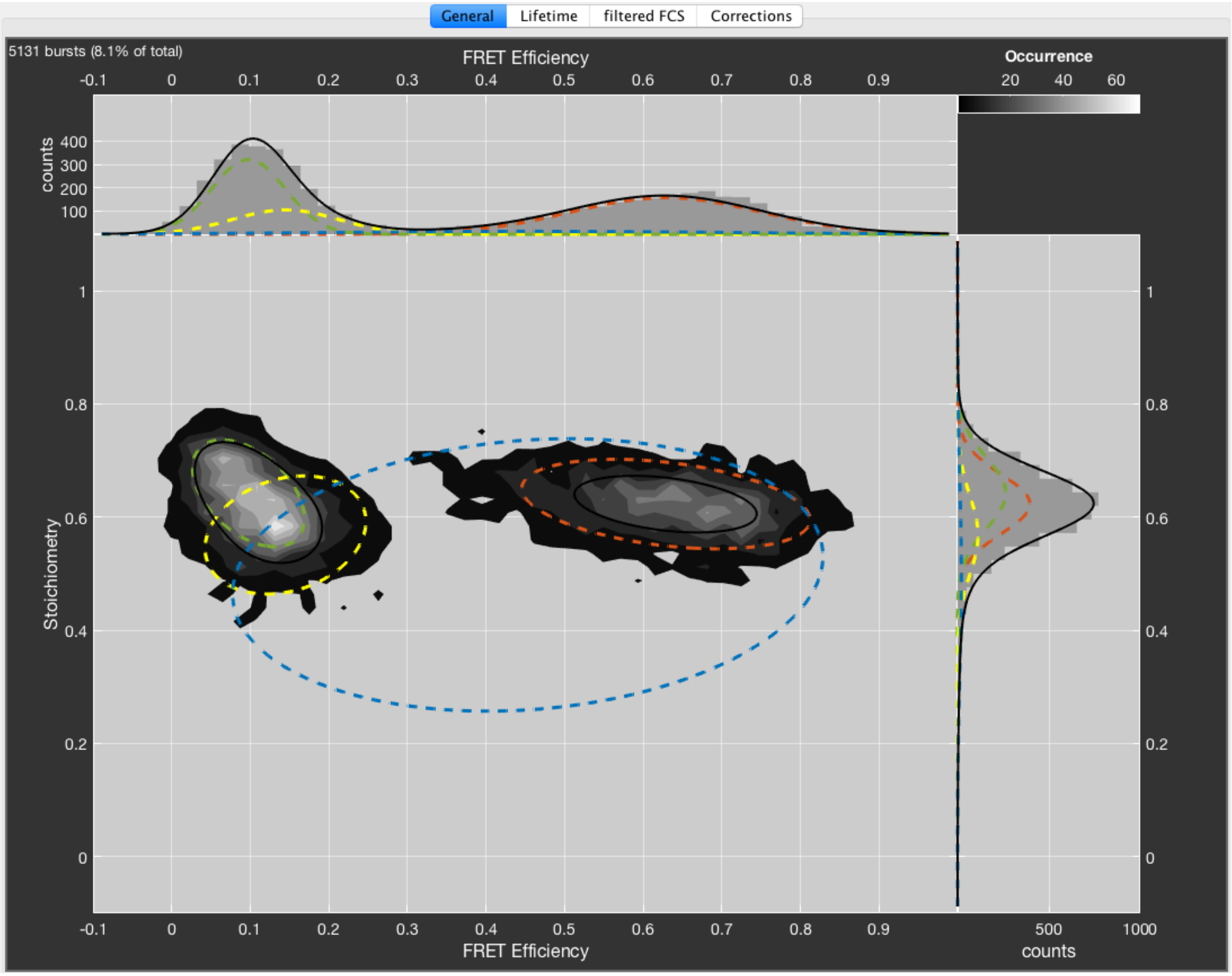


Fig. 20: Example result of Gaussian mixture fitting.

Note: The ellipses for the individual populations are isolines of the distributions. You can specify the isoline height using the *Isoline height for Gaussian fit display* options between 0 and 1. The colors for the populations are given by the *Line Colors*.

Hint: To easily export the fit result, *right-click* the table and select *Copy fit result to clipboard* to transfer the data to a spreadsheet.

8.5.1 MLE fitting

Fitting of Gaussian distributions using maximum likelihood estimation is implemented using the **fitgmdist** function of MATLAB. The algorithm operates on the raw data and is thus independent of the used binning for the 2D plot. MLE fitting does not allow for fixing of parameters or restriction via lower and upper boundaries. The MLE fit additionally returns information on convergence (0 if the fit did not converge, 1 otherwise), the negative logarithm of the likelihood at the found minimum (*-logL*) and the Bayesian information criterion (*BIC*), defined as:

$$BIC = -2\ln\mathcal{L} + \ln(n)k$$

Here, $\mathcal{L}$ is the likelihood of the model, n is the number of data points and k is the number of free fit parameters. Use the *BIC* to decide how many distributions are needed to describe the data; the model with the lowest value for the *BIC* should be preferred. See the wikipedia page for more information on the *BIC*.

8.5.2 LSQ fitting

The least-squares fitting routine minimizes the squared error between data and model using the **lsqcurvefit** function of MATLAB. You can fix parameters (**F**)and set their lower and upper boundaries (**LB** and **UB**) in the fit parameter table. The LSQ fit will return the reduced chi-squared (**red. Chi2**, $\chi^2_{\mathbf{red}}$) as a measure for the goodness-of-fit, defined by:

$$\chi^2_{\mathbf{red}} = \sum_{x,y} \frac{(D(x,y) - M(x,y))^2}{D(x,y)}$$

Here, the $\chi^2_{\mathbf{red}}$ is calculated assuming Poissonian statistics for estimating the uncertainty in the data. Note that while the $\chi^2_{\mathbf{red}}$ is reported, the fit routine is solely based on the squared error and not the $\chi^2_{\mathbf{red}}$.

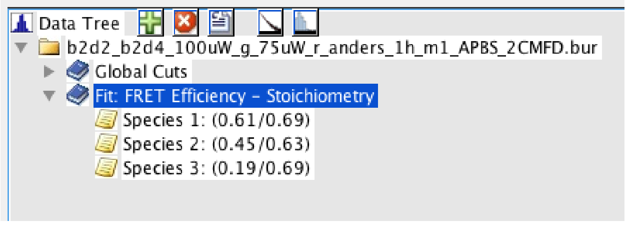
8.5.3 1D parameter fit

Single parameter fits using the above described methods can be performed by simply selecting the same parameter for x- and y-coordinates.

Fitting of FRET efficiency histograms with customizable model functions is additionally implemented in *FCSFit*.

8.5.4 Defining species from fit result

The obtained fit result can be used to sort bursts into species. Bursts belonging to a certain bin in the 2D plot will be assigned to the different fit species based on the relative contributions of the fit species. This functionality is accessed by *right-clicking* the fit result table and selecting **Define species from fit**. A new selection will be added to the **species list**, encompassing all selected bursts used for the fit.



Data Tree

b2d2_b2d4_100uW_g_75uW_r_anders_1h_m1_APBS_2CMFD.bur

Global Cuts

Fit: FRET Efficiency - Stoichiometry

Species 1: (0.61/0.69)

Species 2: (0.45/0.63)

Species 3: (0.19/0.69)

Fig. 21: Burst selection defined from fit result. Species 1 was defined from the fitted component with mean values for FRET efficiency = 0.61 and Stoichiometry = 0.69.

The name for the parent species contains the used parameters for the fit. The subspecies names contain the center values of the respective parameters that were used to define the species.

Hint: This functionality can in principle be used to remove "background" species from the analysis. However, assignment of "background" signal needs to be justified by control measurements. If, for example, the used buffer contains fluorescent contaminations, a burst analysis on buffer-only can identify the background.

8.6 FCS analysis

8.6.1 Burst-wise FCS

Burst-wise FCS can elucidate differences in diffusion time and thus interactions or conformational dynamics on the sub-millisecond timescale. When calculating correlation functions on time scales similar to the length of the recorded signal, as is the case for burst-wise FCS, sampling artifacts need to be considered. In other words, long time lags e.g. on the timescale of diffusion are sampled less frequently in burst-wise time traces than short time lags. For a given time lag τ and burst duration T, the time lag can only be sampled for photons detected in the range $[0, T - \tau]$. In practice, individual correlation functions are calculated for every contributing burst and subsequently averaged while taking care of the uneven sampling:

$$G_{\mathbf{bw}}(\tau) = \frac{\sum_k n_k(\Delta t = \tau) \sum_k (T_k - \tau)}{\sum_k n_k(t \leq T_k - \tau) \sum_k n_k(t \geq \tau)}$$

Here, the sums go over the number of bursts k, $n_k(\Delta t = \tau)$ is the number of photon pairs in burst k with time lag τ, T_k is the burst duration, and $n_k(t \leq T_k - \tau)$ and $n_k(t \geq \tau)$ are the number of photons in the interval $[0, T_k - \tau]$ and $[\tau, 0]$.

Burst-wise FCS can detect conformational dynamics on timescales faster than the diffusion. The diffusion time determined by burst-wise FCS, however, is strongly dependant on the chosen burst search parameters, i.e. higher photon count thresholds will terminate the burst earlier, leading to shorter burst durations. This effect can be diminished by adding a time window around the bursts, enabling a more accurate determination of the diffusion part of the FCS curve. Another effect of the burst selection is that regions of high count rate are selected from the measurements for correlation, resulting in reduced fluctuations and thus lower FCS amplitudes since the average intensity is overestimated. Additionally, since correlating regions are selected, the correlation function usually does not diminish to 0 at large time lags, requiring a significant offset to be considered in the model function.

The figure below illustrates the effect of the chosen time window on the shape and amplitude of the obtained burst-wise FCS curves. A larger time window results in higher amplitudes of the correlation function. The burst-wise FCS curve underestimates the diffusion time, which is asymptotically increased with increasing time window. Obviously, also the maximum time lag scales with the chosen time window. Since the time window is added both to the start and end of the bursts, the maximum lag time will be approximately twice the time window size.

See [Laurence2007] for more information on burst-wise FCS.

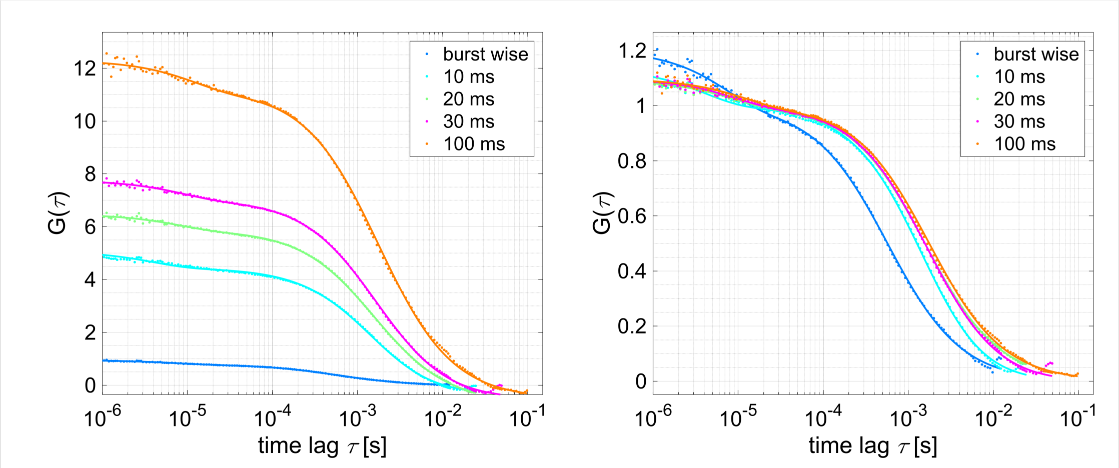
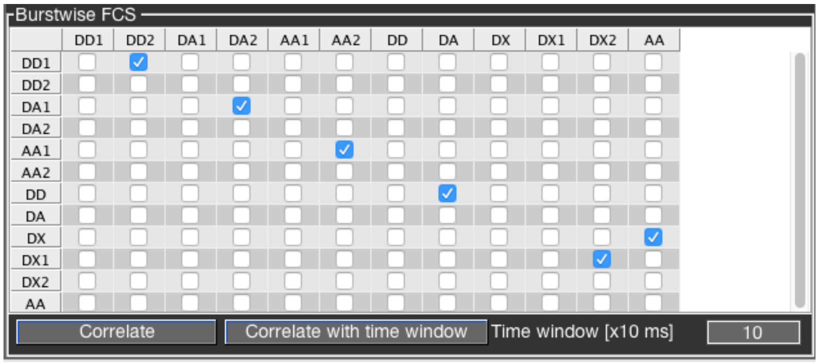


Fig. 22: Burst-wise correlation with increasing time window, unnormalized (left) and normalized to the fitted number of particles (right). For all curves the offset has been subtracted.

When adding a time window, it is possible that adjacent single molecule events fall into the region around the respective burst, leading to contamination of the species-selective correlation function. If another single molecule event that was detected by the burst search falls into the time window around a contributing burst, the burst is not included in the correlation function. This will ensure that the obtained correlation function is specific to the selected species. However, contribution of contaminating signal that was not detected as burst will not be filtered out.

Note: Generally, for burst-wise FCS, it is recommended to use low intensity thresholds for the burst search to detect as much of the fluorescence signal of the single molecules as possible, smoothing the edges of the burst-wise correlation function. Additionally, if species-selective FCS analysis is to be performed, the concentration should be chosen sufficiently low to leave enough time between individual events.

8.6.2 Using the burst-wise FCS module



	DD1	DD2	DA1	DA2	AA1	AA2	DD	DA	DX	DX1	DX2	AA
DD1	□	☑	□	□	□	□	□	□	□	□	□	□
DD2	□	□	□	□	□	□	□	□	□	□	□	□
DA1	□	□	□	☑	□	□	□	□	□	□	□	□
DA2	□	□	□	□	□	□	□	□	□	□	□	□
AA1	□	□	□	□	□	☑	□	□	□	□	□	□
AA2	□	□	□	□	□	□	□	□	□	□	□	□
DD	□	□	□	□	□	□	□	☑	□	□	□	□
DA	□	□	□	□	□	□	□	□	□	□	□	□
DX	□	□	□	□	□	□	□	□	□	□	□	☑
DX1	□	□	□	□	□	□	□	□	□	□	☑	□
DX2	□	□	□	□	□	□	□	□	□	□	□	□
AA	□	□	□	□	□	□	□	□	□	□	□	□

Time window [x10 ms] 10

Fig. 23: The burst-wise FCS GUI in the Corrections/FCS tab.

To perform burst-wise FCS analysis, **BurstBrowser** pre-defines combinations of PIE channels, namely the channels *DD* for donor fluorescence after donor excitation, *DA* for acceptor fluorescence after donor excitation, *AA* for acceptor fluorescence after acceptor excitation and *DX* = *DD* + *DA*, i.e. all photons after donor excitation. Parallel and perpendicular polarization channels are indicated by number (1 is parallel, 2 is perpendicular), thus *DX1* = *DD1* + *DA1*. When no number is given, the channel combines both polarizations, e.g. *DD* = *DD1* + *DD2*.

FRET-FCCS analysis requires the autocorrelations and the cross-correlation of the donor and FRET signals, i.e. in terms of PIE channels the correlation functions for *DD1 x DD2*, *DA1 x DA2* and *DD x DA*. Note that for the autocorrelations, channels with different polarization should be cross-correlated to avoid detector artifacts like dead time or afterpulsing. On the other hand, for the cross-correlation the polarization channels should be combined, i.e. *(DD1+DA2) x (DD2+DA2)*, to ensure that all photon information is used. Other relevant FCS curves for FRET-FCCS analysis are the autocorrelation of the acceptor signal after acceptor excitation *AA1 x AA2*, as well as the autocorrelation of the total signal after green excitation *DX1 x DX2*, which will not be sensitive to FRET dynamics.

A default selection of these correlation functions can be set by *right-clicking* the table and selecting **FRET FCCS selection**. Likewise, the selection in the table can be reset by clicking **Reset**. The divider coarse-grains the time resolution of photon arrivals; its function is describe in *this section*.

Click the **Correlate** button to perform burst-wise FCS without any time window, and the **Correlate with time window** button for correlation with time window. The time window is specified in units of 1 ms. The chosen time window will be added before and after the burst, resulting in a total signal trace length of twice the time window. A time window can only be included if the **Save total photon stream** option was selected for the burst analysis. See the *burst search section* in the **PAM** manual for more details.

8.6.3 Burst-wise diffusion time and diffusion coefficients

The burst-wise correlation can also be used to determine a burst-wise diffusion time by fitting of the individual correlation functions for every single molecule event. To use all available signal, the correlation function is computed from all photons belonging to a burst (i.e. GG1+GG2+GR1+GR2+RR1+RR2). Any channel selections in the correlation table are thus disregarded. Since the burst-wise correlation functions are undersampled and thus noisy due to the limited number of photons per burst, a simple 2D diffusion model is used to fit the burst-wise diffusion time τ_D:

$$G(\tau) = \left(1 + \frac{\tau}{\tau_D}\right)^{-1}$$

Correlation functions are fitted in the rang of 10 μs up to one tenth of the signal trace length. The amplitude parameter is eliminated from the fit procedure by normalizing the correlation function to the average of the first 10 data points. Burst-wise fitting of correlation functions only yields useful results when adding a time window around the bursts. When used only on the burst signal trace, diffusion times are underestimated. The diffusion coefficient is calculated by:

$$D = \frac{\omega_r^2}{4\tau_D}$$

To access this functionality, *right-click* the **Correlate with time window** button and select *Fit burst-wise diffusion time*. The functionality is only available with the usage of time windows around the burst

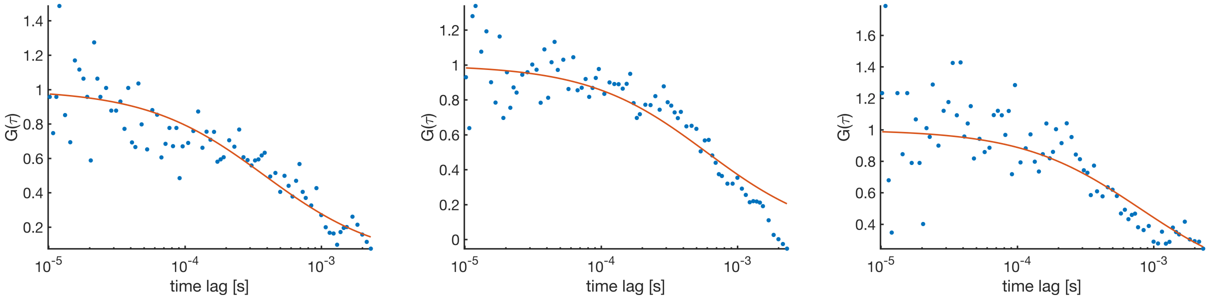


Fig. 24: Examples of burst-wise correlation functions and fits. Time window length is 10 ms.

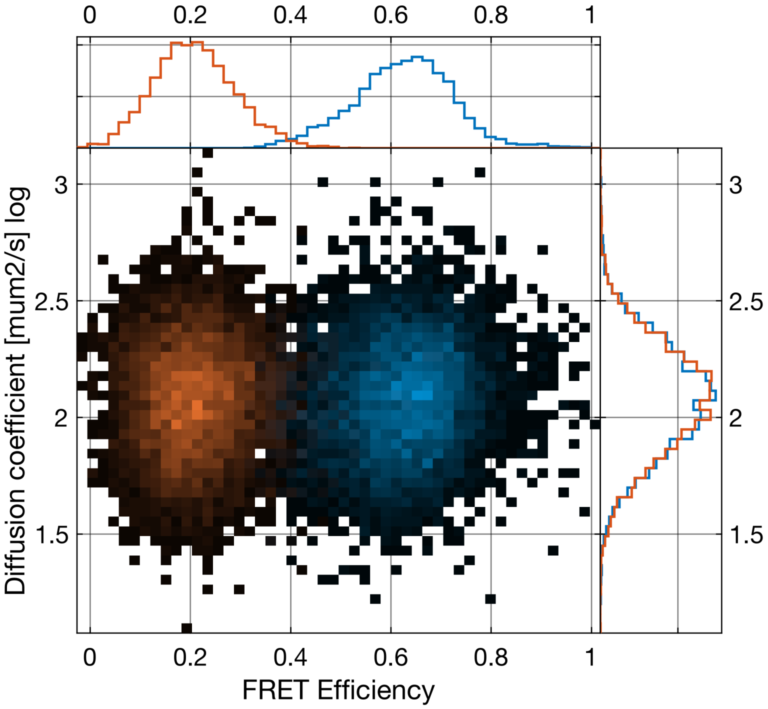


Fig. 25: Example of burst-wise diffusion coefficient analysis. Shown is a measurement of two DNA molecules with identical structure, but different labeling position, showing different FRET efficiencies but identical diffusion coefficient distributions (shown is the logarithm of the diffusion coefficient).

Note: Choose a large enough time window (> 5 ms) to ensure that the diffusion coefficient is not overestimated due to the burst-search algorithm.

8.6.4 Using the filtered-FCS module

Filtered FCS is an extension of fluorescence lifetime correlation spectroscopy (FLCS), extending the theory to include other parameters in addition to the fluorescence lifetime that are available in MFD, e.g. the fluorescence anisotropy and color information (i.e. FRET efficiency). See the *corresponding section* in the **PAM** manual.

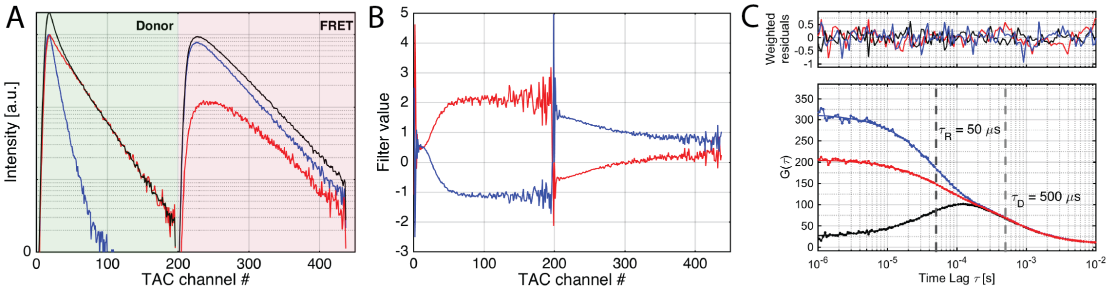
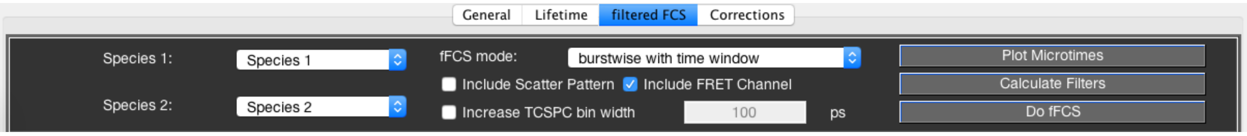


Fig. 26: An example of filtered FCS analysis. (A) Stacked microtime histograms of the measurement of the mixture (left) and the high and low FRET efficiency species (blue and red). Donor and FRET-sensitized acceptor microtime histograms are stacked to use the color information in filter generation. (B) Corresponding statistical filters. (C) Resulting species auto- and cross-correlation functions.

To resolve dynamics in the sub-nanosecond range, fFCS requires cross-correlation between two detectors to avoid dead time and other electronic artifacts. fFCS is thus only implemented for setups using two detection channels per color, either split by polarization or using a 50:50 beam splitter. Separate filters are generated on the two independent detection channels, which are cross-correlated when calculating the correlation functions.

Filtered-FCS functionality is integrated into **BurstBrowser**, allowing the use of subsets of bursts to define the filters for the pure species.



Species 1: Species 1

Species 2: Species 2

fFCS mode: burstwise with time window

☐ Include Scatter Pattern ☑ Include FRET Channel

☐ Increase TCSPC bin width 100 ps

Fig. 27: Filtered-FCS options panel

The filtered-FCS module requires the definition of a parent species in the **Species List** with at least two child species, representing different subsets of the parent species. **BurstBrowser** will then treat the parent species as the *total measurement*, which will be described as a superposition of the sub-species.

The functionality of the filtered-FCS module will be demonstrated using the data set shown in the following figure. In the measurement, fast fluctuations between low and high FRET efficiency states are observed on the μs timescale, as indicated by the deviation from the static FRET line.

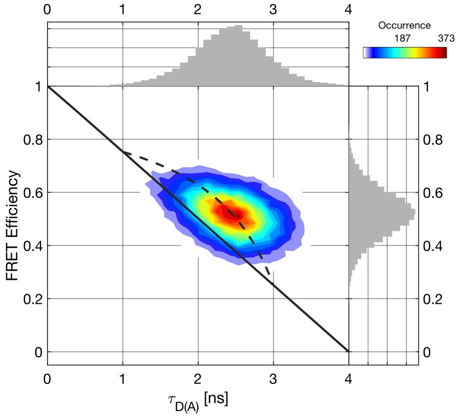


Fig. 28: The E vs. $\tau_{D(A)}$ plot indicates fast dynamics on the sub-millisecond timescale.

To perform the filtered-FCS analysis, we define two sub-species, representing the low and high FRET efficiency states by using thresholds of $E < 0.35$ and $E > 0.75$, respectively (*Species 1* and *Species 2*). Click **Plot Microtimes** to plot the microtime patterns of the parent species (black) and *Species 1* (green) and *Species 2* (blue). Press the **Calculate Filters** button to define the filters for the two species. You can inspect the filters for the parallel and perpendicular channels in the plots at the bottom of the window. In the *Reconstruction* tab, inspect the measured (black) microtime pattern and the microtime pattern reconstructed from the decomposition (red) to judge the quality of the generated filters. Look for good agreement between the two patterns. If the two patterns do not agree well, usually a species that is present in the total measurement is not accounted for by any of the defined species (e.g. no background/scatter species included).

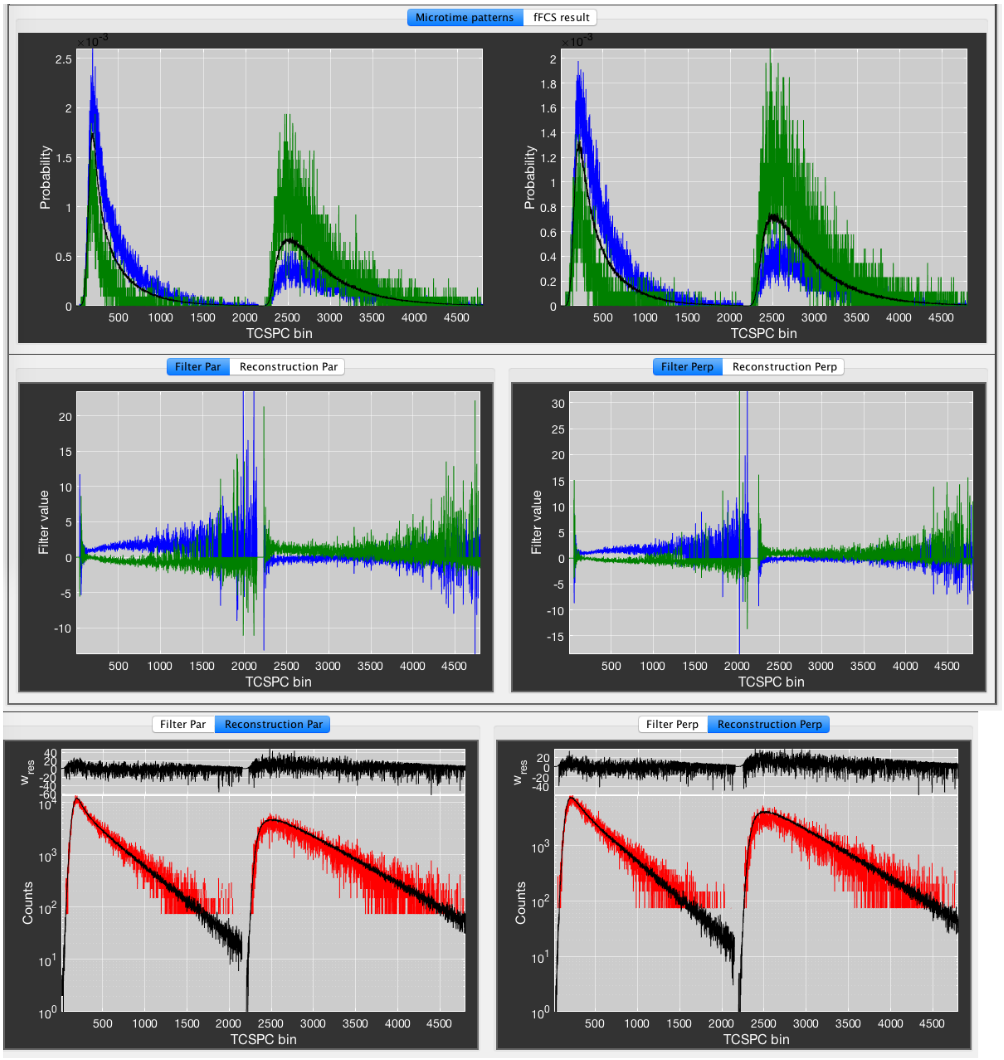


Fig. 29: An example for fFCS analysis of the data in the previous figure. Blue species: E < 0.35, green species: E > 0.75.

Finally, click **Do fFCS** to perform the correlation analysis. The fFCS result will be plotted in the **fFCS results** tab. It will not be saved automatically. To save the result, click the **Save result** button to store the two autocorrelation and two cross-correlation functions in the folder of the *.bur* file.

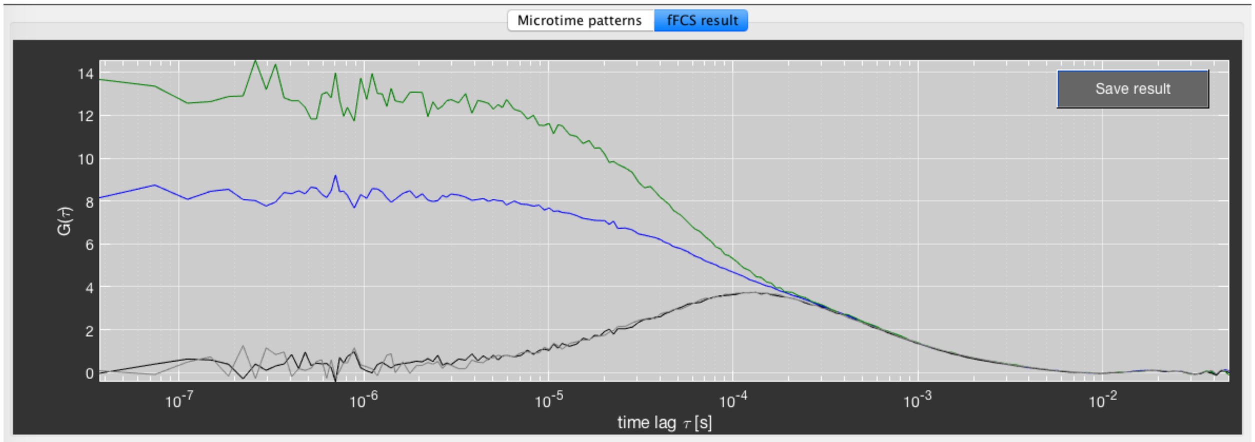


Fig. 30: The result for the exemplary fFCS analysis.

There are several options to the fFCS analysis.

The **fFCS mode** allows you to select among the following approaches:

burstwise: This mode will only use photons belonging to the selected bursts, identical to performing burstwise FCS analysis without the addition of a time window.

burstwise with time window: This will add a time window around the selected bursts, identical to performing burstwise FCS analysis with the addition of a time window. It will use the time window specified in the FCS submodule. Since the addition of a time window will add more background signal, it is advised to include the scatter pattern to address the background contribution. An *.aps* file for the measurement, generated by checking the *Save Total Photon Stream* option upon performing of the burst analysis, is required.

continuous photon stream (with donor only): This option will use the total measurement for the filtered-FCS analysis. As a consequence, all species present in the measurement need to be accounted for. Use the option without inclusion of a donor-only species only for samples that contain no or negligible amount of donor-only molecules. If donor-only molecules are present, usage of the option … *with donor-only* will determine the microtime pattern from the burst measurement using a stoichiometry threshold and include it in the analysis. Since the filtered-FCS analysis is not performed on the channel of acceptor emission after acceptor excitation (*AA*), there is no need to account for acceptor-only molecules. An *.aps* file, generated by checking the *Save Total Photon Stream* option upon performing of the burst analysis, is required. Since most of the signal will be originating from background, the inclusion of the scatter pattern as a species is required.

The option **Include Scatter** adds the stored scatter/background pattern as a third species. Enable this option when using the total measurement or including a large time window around the bursts.

Check **Include FRET channel** to stack the microtime histogram of the FRET channel with the donor channel. This increases the statistics by including FRET-induced acceptor photons, especially if very high FRET species are present.

To deal with noise microtime patterns cause by low statistics, e.g. if only a few bursts can be used to define a species, you can **increase the TCSPC bin width**, as is illustrated in the following figure. The reduced microtime resolution is only used for the filter calculation and does not affect the data in any other way.

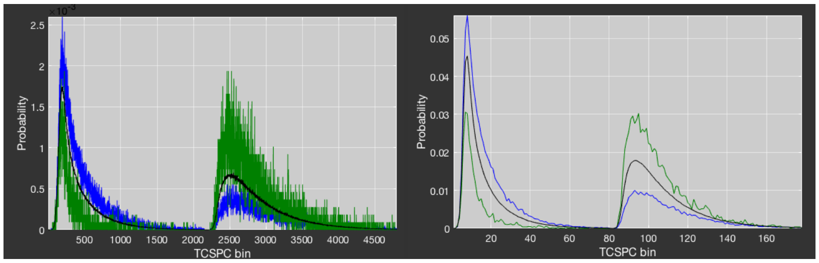


Fig. 31: An example for increasing the bin width to deal with noisy microtime patterns.

Using synthetic microtime patterns

Another way to deal with statistics for the microtime patterns, or to include external microtime patterns, is to fit the fluorescence decay using **TauFit** and *export the fitted microtime pattern*. Note that it is required to use an anisotropy fit model to generate synthetic microtime patterns for both parallel and perpendicular decay. The usage of the FRET channel is not supported with synthetic microtime patterns, since there is currently no fit model for donor and acceptor decay simultaneously.

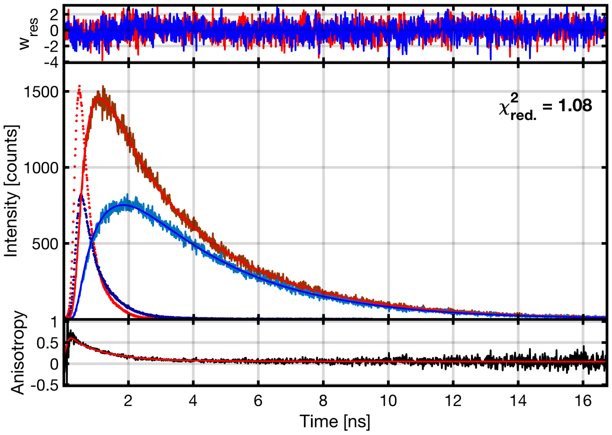


Fig. 32: Synthetic microtime pattern generation in **TauFit** using an anisotropy fit model.

To load the *.mi* files, click the popupmenu of a species and select *Load synthetic pattern…*. Usage of synthetic patterns increases the quality of the filters through the reduced noise.

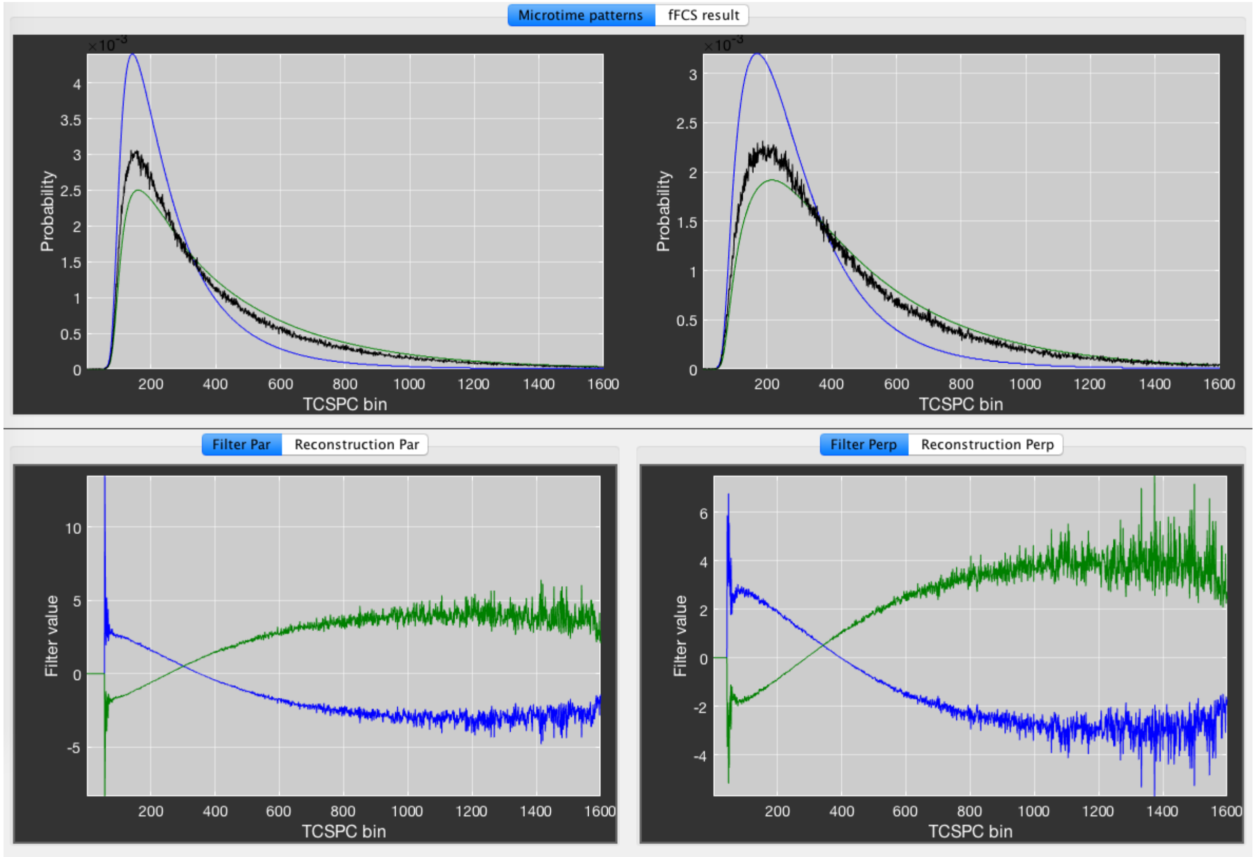


Fig. 33: Example for using synthetic microtime patterns to generate filters for fFCS analysis.

Additionally, if the pure species are available in separate measurements, you can use the *filtered-FCS module* in **PAM** to export them as *.mi* files that can be loaded in **BurstBrowser**.

8.7 Exporting plots and data

8.7.1 Choosing the export location

By default, **BurstBrowser** saves exported files into the folder of the respective *.bur* file. If you are analyzing multiple data sets, it is convenient to collect all exported plots and data into a single folder. Disable the default behavior by unchecking the menu option *Export -> Export to Current File Path*. Choose a custom export path through *Export -> Change Export Path*. You can check the currently set export path under *Export -> Current Export Path*.

Note: The *Export Path* affects the save location of all exported plots and exported FRET histogram data. It does not affect the save location of PDA data or microtime patterns.

8.7.2 Exporting plots

Export of plots is implemented for all plots in the *General* and *Lifetime* tab, and are accessed via *right-clicking* the respective plots.

In the *General* tab, one-dimensional or two-dimensional plots can be exported individually. The *Lifetime* tab allows to export all lifetime-related plots in one figure (*Export Lifetime Plots* in the *All* tab), or the individual two-dimensional plots including the marginal parameter distributions in the *Individual* tab.

The resulting figures can be saved manually as any image format supported by MATLAB. Alternatively, you can have the program prompt you for saving when closing the figure by checking the *Save file when exporting figure* checkbox in the *Options* tab. The default output format is *.png*.

The *Export all graphs* button in the *Options* tab will automatically export the following graphs for the selected species: FRET efficiency vs. Stoichiometry, FRET efficiency, all lifetime-related plots and FRET efficiency vs donor lifetime.

8.7.3 Exporting data

FRET efficiency histogram export

FRET efficiency histograms can be exported to compare at a later time point or for further analysis in **FCSFit**. Exporting can be performed by *right-clicking** the species in the **Species List** and selecting *Export -> Export FRET Efficiency Histogram*, which will save a *.his* file containing the FRET efficiency information of the selected bursts into the specified export folder. To perform export actions for multiple species at once, select the respective species and select the menu *Export -> Export FRET Efficiency Histograms … -> for all selected species*. Selecting *for all selected files* will instead export the last selected species for all files.

If the FRET efficiency histogram changes shape over the course of the measurement, you can export a *time series* to extract the kinetics from the data. *Right-click* the species in the **Species List** and select *Export -> Export FRET Efficiency Histogram (Time Series)*. Specify the time window size in the following edit box. This will export FRET efficiency histogram of the measurement for non-overlapping time windows of the given size.

To compare previously exported *.his* files, select the menu *Compare -> Compare FRET efficiency histograms from .his files* and select the *.his* files you want to compare.

To globally analyze FRET efficiency histograms, load the *.his* files in **FCSFit** as described in the *respective section*.

Photon distribution analysis export

To analyze FRET efficiency data more quantitatively, it can be exported to *PDAFit*. The menu structure and export functionality is equivalent to FRET efficiency histogram export as described in the *previous section*. In addition to the menus, PDA export can be accessed by clicking the histogram button in the top row of the **Species List**.

PDA requires equivalent time bins for the FRET efficiency histogram analysis. The time bin size is specified in the *Options* tab in the *Data Processing* panel (default value: 1 ms). For kinetic analysis, it might be of interest to perform global analysis on the same data set exported with different time bin lengths. The edit box for specifying the time bin accepts standard MATLAB expressions to specify a list of time bins to export. Comma-separated lists of time bins (e.g `0.5,1,1.5,2`) or range expression (e.g. `0.5:0.5:2`) are accepted.

PDA data will always be exported into the folder of the respective *.bur* file and has the ending *.pda*.

Exporting microtime data

Microtime data of selected species can be exported by *right-clicking* the species and selecting *Export -> Export Microtime Pattern*. Exported microtime pattern (*.mi* extension) are used in **PAM**'s *filtered FCS module* to define pure species. If the patterns are too noisy due to low photon statistics, it is advisable to fit the microtime hisograms and use the fitted model function instead of the raw data. See the *respective section* in the **TauFit** manual for details.

Exporting to TauFit

Sub-ensemble TCSPC analysis on selected species can be performed by *right-clicking* and selecting *Send selected Species to TauFit*, or by selecting the fluorescence decay symbol in the top of the **Species List**. This will export the cumulative TCSPC histogram to *TauFit*. Note that data transfer is only implemented for single species and not integrated with multiselection. See the section on *sub-ensemble TCSPC* for more details.

8.8 Additional functionalities

8.8.1 Time window analysis

Dynamic interconversion between several FRET efficiency states on the sub-diffusion timescale will lead to averaging of the observed FRET efficiency dependant on the observation time. *Right-click* the species and select *Time Window Analysis*. This will plot the FRET efficiency histogram for time windows from 0.5 to 3 ms in intervals of 0.5 ms. Specify the minimum number of photons per time window to exclude time windows with low amount of photons and thus reduce shot noise in the histograms.

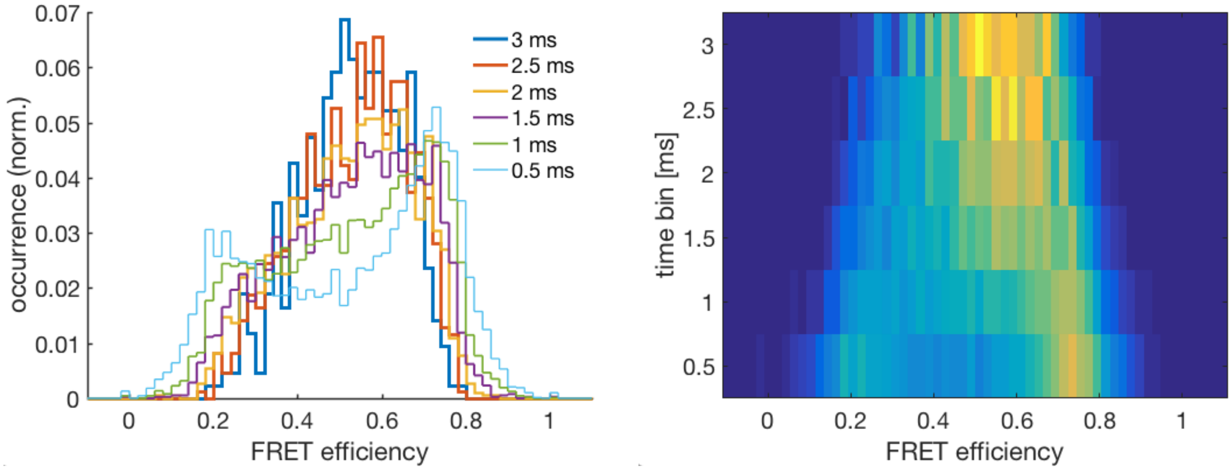


Fig. 34: An example for identifying dynamics using time window analysis. The two distinct populations at low time windows are averaged at longer time windows.

8.8.2 Burst variance analysis

Given a FRET distribution broader than the expected shot-noise distribution, Burst variance analysis (BVA) can be used to distinguish between multiple static components or multiple interconverting states. *Right-click* the species and select *Burst Variance Analysis*. This will plot the observed and expected standard deviation of the proximity ratio for each burst and the corresponding confidence interval (CI) against the selected X-axis. Each burst will be segmented into consecutive windows of a specified number of photons. The standard deviation is binned using a width of 0.05 along the X-axis. Specify a threshold for the number of bursts per bin to increase statistical power by excluding bins with a low amount of bursts. The CI considers the sampling distribution of standard deviations expected for the number of photon windows by using a Monte Carlo approach. Specify the sampling number (i.e. number of simulations) to get a binomial distribution of standard deviations (e.g. 100 for a valid CI or 1 for a fast calculation). Specify the CI α to adjust the upper-tail CI (i.e. low α value for higher confidence interval and high α value for lower CI). Select either the proximity ratio or FRET efficiency as the X-axis (note: for FRET efficiency the expected standard deviation has an oval shape).

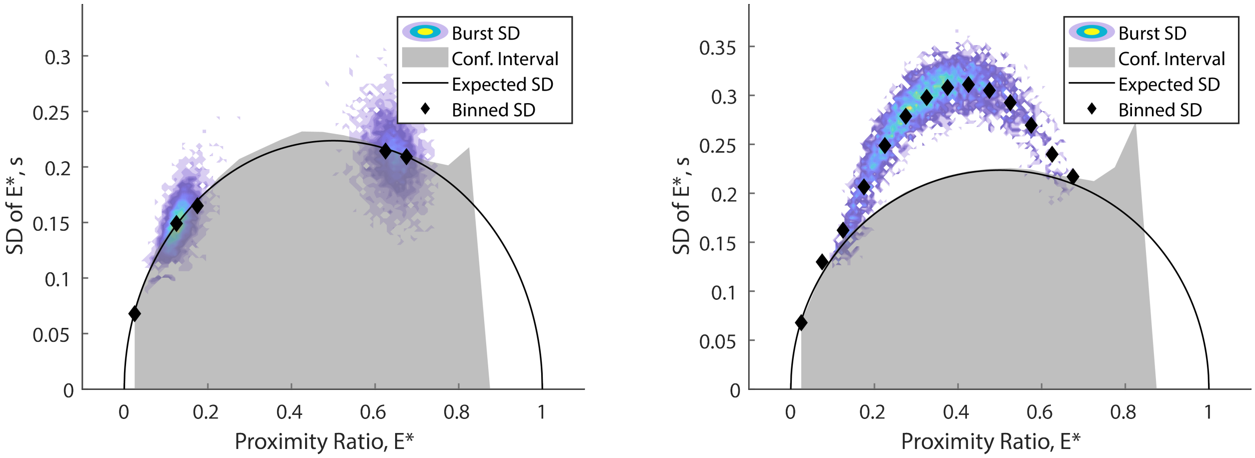


Fig. 35: An example for identifying dynamics using burst variance analysis. Standard deviation values clustering around the expected standard deviation indicate static FRET (left panel) whereas bursts with standard deviation values significantly above the confidence interval indicate within-burst dynamics (right panel).

8.8.3 Working with large data sets

BurstBrowser offers functionality to work with multiple files simultaneously, such as synchronizing *correction factors* and *cut states*. Additionally, many *export functionalities* work with multiselection.

To load multiple measurements from a single folder, select *File -> Load New Burst Data* (shortcut *ctrl+N*) and select the measurements to load. Add burst files to the currently loaded files by selecting *File -> Add burst data*. To easily load many measurements at once, use the *File -> Load New Burst Files from Subfolders* menu. This will search through all subfolders of the selected folder and load any *.bur* files found.

To keep track of files that belong together, you can group them into a database. In the *Database* tab, select *Add files to database* to add new files to the current database. Use *Append loaded files to database* to add all loaded to the current database. *Create database from folder* has the same functionality as the option to load burst files from subfolders. Individual files can be removed from the database by selecting them in the list and using the *del* key. To load a selection of files from the database, use *enter*. Databases can be saved as *.bdb* files and reloaded at a later time point. Additionally, **BurstBrowser** remembers the last used database, so you can continue working on the last dataset immediately.

In addition, **BurstBrowser** keeps track of the last 20 loaded files in the *File History*. Use *enter* to load selected files from the *File History*, and *del* to remove files.

8.8.4 Saving the analysis state

Save the edits to all currently loaded files by selecting the menu *File -> Save Analysis State* (shortcut *ctrl+S*). This will save the cut state and corrections/parameters. Any fit lines in the lifetime plots, correction plots or the state of filtered-FCS analysis will however not be remembered.

Hint: To avoid loss of data, enable the *Ask for saving when closing program* checkbox in the *Options* tab. This will prompt you to save the analysis state when closing **BurstBrowser**. Select *Discard* to close the program without saving the changes, and *Cancel* to abort the closing progress and continue with the analysis.

8.8.5 Keeping notes

BurstBrowser offers an internal *Notepad* which you can use to keep temporary notes. Open it using the menu *Notepad* or by shortcut *ctrl+T*. The notes are saved in the currently selected **PAM** profile and kept between sessions. You can save the contents of your current notepad as a *.txt* file by *right-clicking* the text and selecting *Save contents*.

8.9 Table of parameters in BurstBrowser

8.9.1 List of correction factors

γ-factor: Accounts for differences in quantum yield Φ and detection efficiency g. Given by

$\gamma = \frac{\Phi_A g_A}{\Phi_D g_D}$

β-factor: Accounts for differences in extinction coefficient ϵ and incident laser power I in alternating excitation at the given excitation wavelength λ_{exc}, as defined in [Lee2005]:

$\beta = \frac{\epsilon_A^{\lambda_{\text{exc}}^A} I_A}{\epsilon_D^{\lambda_{\text{exc}}^D} I_D}$.

crosstalk/ct: Fraction of donor signal that is detected in the acceptor channel.

direct excitation/de: Fraction of acceptor signal originating from direct excitation of the acceptor by the donor excitation laser. This correction is performed based on the signal of the acceptor after acceptor excitation (RR) and is thus dependent on the acceptor laser power.

G-factor: Corrects differences in detection efficiencies g of parallel and perpendicular polarization. Given by

$G = \frac{g_\perp}{g_\parallel}$

l_1/l_2: Factors accounting for polarization mixing caused by the high N.A. objective lens. See definition of anisotropy.

BG: Background count rate of the given channel in kHz. Photon counts T are corrected for background contribution based on the burst duration T by

$N_{\text{cor}} = N - BG * T$

8.9.2 List of burst-wise parameters

FRET efficiency: Corrected FRET efficiency calculated using

$E = \frac{GR - ct*GG - de*RR}{\gamma*GG + GR - ct*GG - de*RR}$

with background-corrected photon counts.

Stoichiometry: Corrected labeling stoichiometry calculated using

$S = \frac{\gamma*GG + GR - ct*GG - de*RR}{\gamma*GG + GR - ct*GG - de*RR + \beta*RR}$

with background-corrected photon counts. β-factor correction is optional and can be disabled using the checkbox (Ref).

Proximity ratio: Uncorrected FRET efficiency based on the raw signal

$PR = \frac{GR}{GG + GR}$

Stoichiometry (raw): Uncorrected labeling stoichiometry based on the raw signal

$$S = \frac{GG+GR}{GG+GR+RR}$$

Anisotropy D/A: Steady-state fluorescence anisotropy of donor or acceptor fluorophore calculated by

$$r_{\mathrm{ss}} = \frac{G*I_{\parallel}-I_{\perp}}{G(1-3*l_2)I_{\parallel}+(2-3*l_1)I_{\perp}}$$

Lifetime D/A [ns]: Burst-wise fluorescence lifetime of donor or acceptor fluorophore determined from a maximum-likelihood estimator using a constant background contribution. See [Maus2001] for details on the MLE.

Mean macrotime [s]: Burst-wise mean macrotime (arrival time) in seconds with respect to the start of the measurement.

Number of photons: Uncorrected number of photons per burst for selected channel.

Duration [ms]: Duration of burst given by the time interval between first and last photon, determined from all photons or the selected channel.

Count rate [kHz]: Burst-wise average count rate in kHz calculated from total number of photons and burst duration.

|TGX-TRR| filter: Difference in mean macrotime between the combined photons after donor excitation (GX) and the photons after acceptor excitation (RR), normalized to the burst duration. This filter is sensitive to bleaching of donor or acceptor dye. See [Kudryavtsev2012] for more details.

|TGG-TGR| filter: Defined analogously to |TGX-TRR| filter, but between donor and FRET channel. This filter is sensitive to dynamic fluctuations between FRET states. See [Kudryavtsev2012] for more details.

ALEX 2CDE filter: This filter sensitive for fluctuations of the signal after donor excitation with respect to the signal after acceptor excitation, and can thus be used to filter donor or acceptor blinking and bleaching events. See [Tomov2012] for details of filter calculation.

FRET 2CDE filter: Similar to the ALEX 2CDE filter, this filter is sensitive to relative fluctuations of the donor and FRET signal, and can thus be used to detect dynamic interconversion between different FRET states. See [Tomov2012] for details of filter calculation.

FRET efficiency (from lifetime): FRET efficiency determined from burst-wise lifetime $\tau_{D,A}$ and specified donor-only lifetime $\tau_{D,0}$ using

$$E = 1 - \frac{\tau_{D,A}}{\tau_{D0}}$$

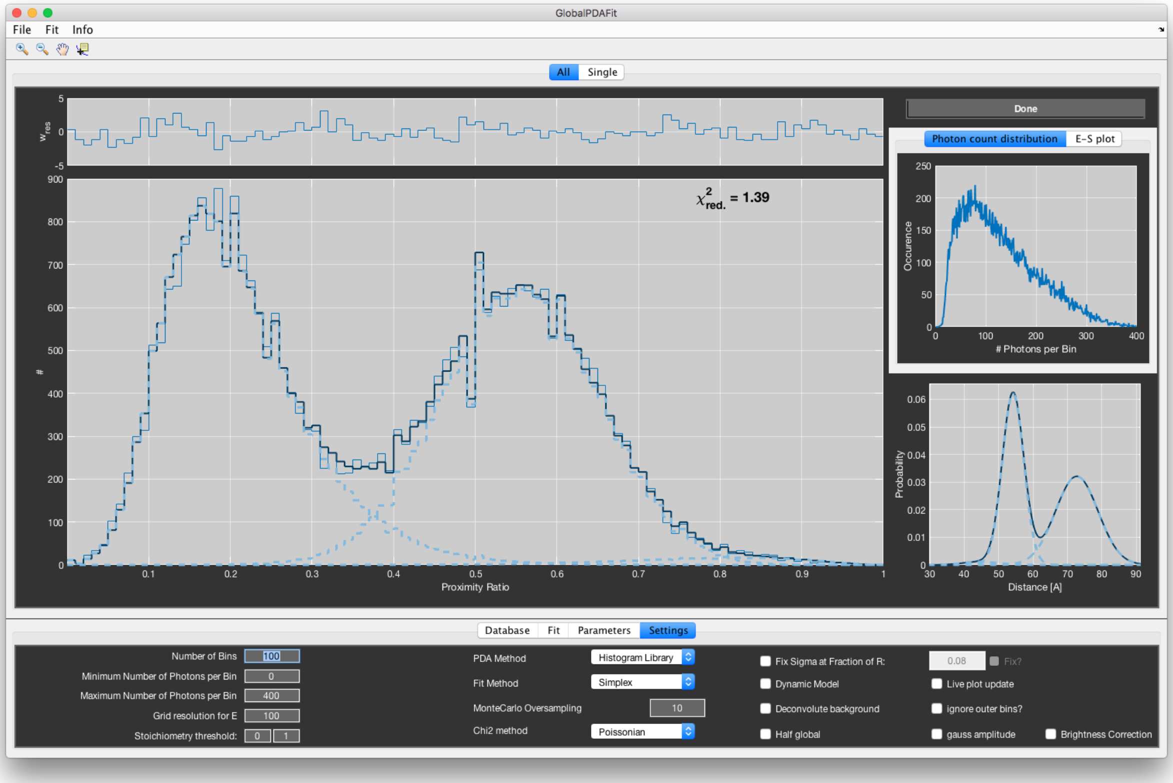
Distance (from intensity/from lifetime) [Angstrom]: Calculated distance from burst-wise FRET efficiency (either determined from intensity or lifetime) using the given Förster radius R_0:

$$R = R_0\left(\frac{1}{E}-1\right)^{-1/6}$$

CHAPTER 9

PDAFit

Contents



Number of Bins 100
Minimum Number of Photons per Bin 0
Maximum Number of Photons per Bin 400
Grid resolution for E 100
Stoichiometry threshold: 0 1
PDA Method Histogram Library
Fit Method Simplex
MonteCarlo Oversampling 10
Chi2 method Poissonian
□ Fix Sigma at Fraction of R: 0.08 □ Fix?
□ Dynamic Model
□ Live plot update
□ Deconvolute background
□ ignore outer bins?
□ Half global
□ gauss amplitude
□ Brightness Correction

Fig. 1: PDAFit GUI

9.1 Photon Distribution Analysis (PDA)

9.1.1 General introduction to PDA

Photon distribution analysis (PDA, [Antonik2006] [Nir2006]) is a statistical method to describe the observed shot-noise limited FRET efficiency histograms by means of the underlying distance heterogeneity. PDA explicitly takes the statistics of photon emission and experimental correction factors into account and achieves higher accuracy than a direct analysis of the FRET efficiency histogram. It is especially useful for identifying heterogeneity in the FRET efficiency distribution that exceeds the expected value from shot-noise alone, allowing to quantify how defined a conformational state is.

PDA targets the uncorrected FRET efficiency histogram (also called proximity ratio histogram, PRH), where the proximity ratio is given by:

$$PR = \frac{N_{GR}}{N_{GG} + N_{GR}}$$

The algorithm simulates an expected proximity ratio histogram based on the input distances, correction factors and observed distribution of total photon counts, which accounts for the shot-noise. By variation of the parameters of the distance distribution, one can find the parameters that best describe the data.

Generally, two approaches have been proposed to simulate the proximity ratio histogram. Seidel and coworkers developed an analytical approach to calculate the PRH [Antonik2006], while Weiss and coworkers used Monte Carlo simulations to generate an approximate PRH [Nir2006]. Both approaches are implemented in the software, together with a related approach based on a maximum likelihood estimator.

Generally, the proximity ratio histogram is given by:

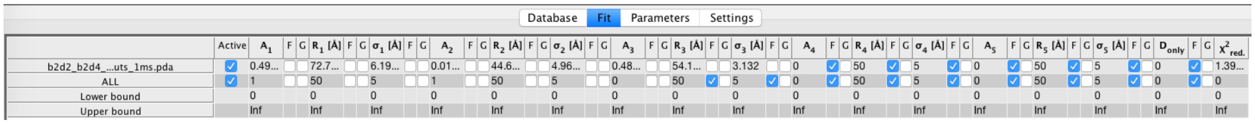
$$P\left(PR \in [a,b]\right) = \sum_{\text{all }(N,N_{GR})\text{ for which }\left(\frac{N_{GR}}{N}\right)\in[a,b]} P(N)P_{\epsilon}(N_{GR}|N)$$

Here $P(N)$ is the photon count distribution, i.e. the probability to detect N photons, and $P_E(N_{GR}|N)$ is the probability to detect N_{GR} red photons out of N total photons, given the apparent FRET efficiency ϵ, described by a binomial distribution:

$$P_{\epsilon}(N_{GR}|N) = \binom{N}{N_{GR}}\epsilon^{N_{GR}}\left(1-\epsilon\right)^{N-N_{GR}}$$

The apparent transfer efficiency ϵ is hereby a function of the correction parameters. Background photon counts are additionally considered as described in the references.

9.1.2 The model function



	Active	A_1	F	G	R_1 [Å]	F	G	σ_1 [Å]	F	G	A_2	F	G	R_2 [Å]	F	G	σ_2 [Å]	F	G
b2d2_b2d4_...uts_1ms.pda	☑	0.49...	☐	☐	72.7...	☐	☐	6.19...	☐	☐	0.01...	☐	☐	44.6...	☐	☐	4.96...	☐	☐
ALL	☑	1	☐	☐	50	☐	☐	5	☐	☐	1	☐	☐	50	☐	☐	5	☐	☐
Lower bound		0			0			0			0			0			0		
Upper bound		Inf			Inf			Inf			Inf			Inf			Inf		

	A_3	F	G	R_3 [Å]	F	G	σ_3 [Å]	F	G	A_4	F	G	R_4 [Å]	F	G	σ_4 [Å]	F	G
b2d2_b2d4_...uts_1ms.pda	0.48...	☐	☐	54.1...	☐	☐	3.132	☐	☐	0	☑	☐	50	☑	☐	5	☑	☐
ALL	0	☐	☐	50	☑	☐	5	☑	☐	0	☑	☐	50	☑	☐	5	☑	☐
Lower bound	0			0			0			0			0			0		
Upper bound	Inf			Inf			Inf			Inf			Inf			Inf		

	A_5	F	G	R_5 [Å]	F	G	σ_5 [Å]	F	G	D_{only}	F	G	$\chi^2_{red.}$
b2d2_b2d4_...uts_1ms.pda	0	☑	☐	50	☑	☐	5	☑	☐	0	☑	☐	1.39...
ALL	0	☑	☐	50	☑	☐	5	☑	☐	0	☑	☐	0
Lower bound	0			0			0			0			0
Upper bound	Inf			Inf			Inf			Inf			Inf

Fig. 2: The fit parameter table

PDAFit currently only supports normal distributions of distances with up to 5 states. The model additionally allows the inclusion of a donor-only population, which will behave differently from a low FRET population since no direct excitation of the acceptor dye can occur. Generally, the model function takes the form of:

$$M = (1 - F_{donor-only})PRH\left(\sum_{i=1}^{5}\frac{F_i}{\sqrt{2\pi}\sigma}\exp-\frac{\left(R-\bar{R}_i\right)^2}{2\sigma^2}\right) + F_{donor-only}PRH_{donor-only}$$

I.e. the donor-only fraction is equivalent to the total percentage of donor-only molecules, while the species fractions F_i yield the fractions among the double-labeled molecules.

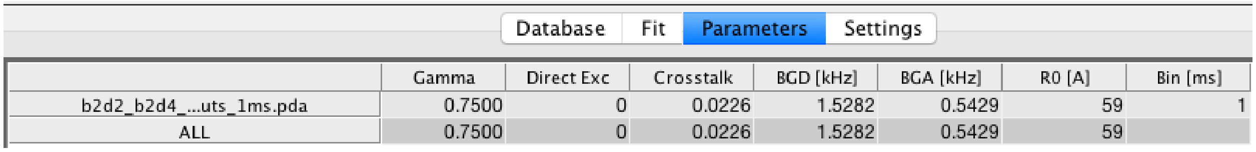
If no conformational heterogeneity is present in the sample and the broadening of the distance distribution is solely due to photophysical artifacts, the observed distribution width σ is proportional to the center distance R [Kalinin2010b]. Select the *Fix Sigma at Fraction of R* checkbox in the *Settings* tab to activate the constraint $\sigma = f * R$, where f is the fraction. You can either leave the fraction as a fit parameter or fix it by checking the *Fix* checkbox next to the edit box.

9.1.3 Generating PDA data

For efficient calculations, PDA requires the data to be of equal time length. Thus, instead of performing the analysis on burst-wise data, the individual bursts are sectioned with a constant time window (typically 1 ms). As a consequence, the obtained proximity ratio histogram in **PDAFit** may look slightly differently from the histogram in **BurstBrowser**.

The process of exporting data from **BurstBrowser** for use in **PDAFit** is described *here*.

9.1.4 The correction parameters



	Gamma	Direct Exc	Crosstalk	BGD [kHz]	BGA [kHz]	R0 [A]	Bin [ms]
b2d2_b2d4_...uts_1ms.pda	0.7500	0	0.0226	1.5282	0.5429	59	1
ALL	0.7500	0	0.0226	1.5282	0.5429	59	

Fig. 3: The parameter table

The correction factors used in **PDAFit** mostly resemble the correction used in **BurstBrowser** (see *correction factors* there). If corrections have been applied in **BurstBrowser**, these are transferred to **PDAFit**. You can inspect the correction factors in the *Parameters* tab.

An exception to this rule is the direct excitation factor. In burst analysis, this correction factor is based on the signal in the direct acceptor excitation channel and thus dependent on the acceptor laser power. However, PDA requires the probability of direct acceptor excitation by the donor excitation laser p_{de}, which is given by:

$$p_{de} = \frac{\epsilon_A^{\lambda_D^{ex}}}{\epsilon_D^{\lambda_D^{ex}} + \epsilon_A^{\lambda_D^{ex}}}$$

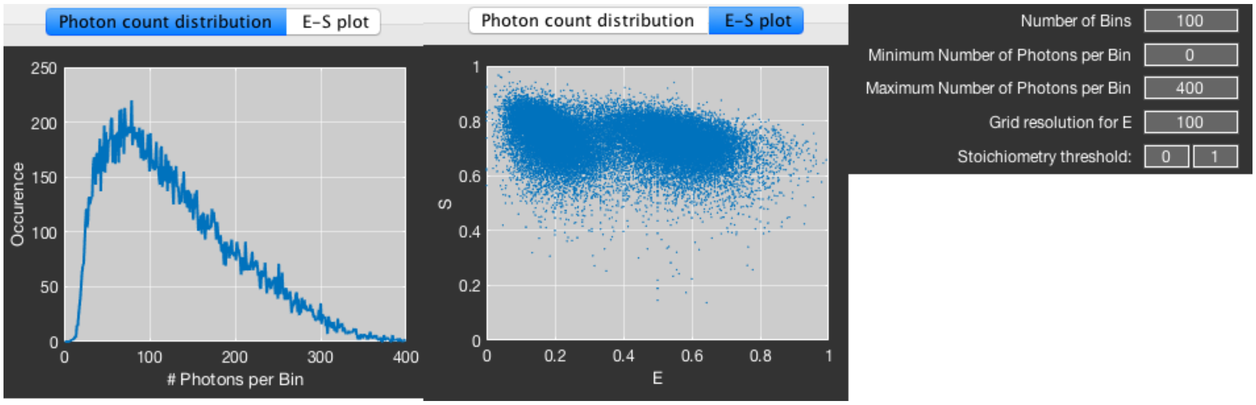
Here, $\epsilon_{D/A}^{\lambda_D^{ex}}$ is the extinction coefficient of the donor or acceptor at the excitation wavelength of the donor fluorophore. This quantity has to be determined for each dye pair and is not accessible from the burst data.

In addition to the correction factors, the bin size of the loaded PDA file is given *Bin [ms]*. This parameter can not be modified. To change the bin size, export the data again from **BurstBrowser** with a different time bin.

When working with multiple files, you can use the *All* row to change the respective parameter for all files.

Note: The calculated histogram library is dependent on the set correction factors. A change in any of the parameters will trigger a recalculation of the histogram library, which can be time consuming. Set the parameters correctly in the beginning to save time.

9.1.5 Data selection



Number of Bins 100

Minimum Number of Photons per Bin 0

Maximum Number of Photons per Bin 400

Grid resolution for E 100

Stoichiometry threshold: 0 1

Fig. 4: Selection parameters

By default, the whole data set is selected. The selection can, however, be restricted in **PDAFit** using the left column in the *Settings* tab.

Minimum number of photons per bin: Set this to a value larger 0 to exclude low photon number time bins from the analysis.

Maximum number of photons per bin: This will limit the maximum number of photons per time bin. The computation time for the histogram method scales with the maximum number of photons per time bin, which makes it beneficial to limit this number, especially if only a few time bins contain a large number of photons, Inspect the photon count distribution to select a reasonable value for this parameter. (The other fit methods do scale mainly with the number of bursts and are mostly independent of the number of photons per time bin.)

Stoichiometry threshold: Additionally, you can limit the allowed stoichiometry range as seen in the Efficiency vs. Stoichiometry plot (E-S). The displayed stoichiometry is calculated based on the raw photon counts $S = \frac{N_{RR}}{N_{GG}+N_{GR}+N_{RR}}$.

Applied selection criteria are globally applied to all loaded data sets.

9.1.6 The *All* and *Single* tab

The *All* tab displays all loaded data sets simultaneously. Select the *Single* tab if you want to inspect a single data set alone. Use the popup menu located in the bottom right of the *Single* tab to select the data set to be displayed. When the *Single* tab is active, only the selected data set will be fit.

9.1.7 Performing a PDA fit

After setting all correction parameters and initial fit parameters, click *Fit -> Start* in the menu. If no global parameter is specified, the program will fit the data sets one after the other. If global parameters are considered, the total chi-squared over all data sets is minimized. At any point, you can stop the fit routine by selecting *Fit -> Stop*. To set good starting parameters, you can preview the histogram based on the currently set model parameters by selecting *Fit -> View*.

Once the fit is finished, you can store the fit result in the data files by selecting *File -> Save to Files(s)*. To obtain figures of the result, select *File -> Export Figures*. This will store the result in a subfolder of the data file.

9.1.8 PDA fit methods

Histogram Library

The *Histogram Library* fit methods implements the PDA method developed by [Antonik2006]. The algorithm generates shot-noise limited histograms for a grid (library) of equally spaced apparent efficiency values over the interval [0,1]. The setting *Grid resolution for E* in the *Settings* tab determines the number of sample points over the interval (set this to at least 100). To implement the distance distribution model, the probability distribution of the distance is transformed into efficiency space. The simulated histogram is then a probability-weighted linear combination generated from the histogram library.

The advantage of this approach is that an accurate histogram is generated. After the initial preparation period, during which the histogram library is generated, the computational load is minimal, since only linear combinations have to be computed.

The accuracy, especially at the extreme edges of very small or large distances, is limited by the resolution of the grid of efficiency values of the histogram library. Especially if the fitted distance distribution show a very narrow width ($< 1\ \mathring{A}$), the grid resolution should be increased. As an example, at vanishing distribution width $\sigma \rightarrow 0\ \mathring{A}$, the algorithm will not find the true value of the center distance, but instead the distance value that belongs to the closest efficiency value of the grid. This effect of the grid, however, is reduced for normal distribution width $\sigma > 2\ \mathring{A}$, where the resulting histogram will be a weighted linear combination of multiple library elements.

MonteCarlo

The *MonteCarlo* method uses the algorithm as proposed by [Nir2006] to simulate the proximity ratio histogram. Hereby, for every data point, a random number generator is used to draw a value for the proximity ratio. To reduce stochastic noise, set the *MonteCarlo Oversampling* parameter in the *Settings* tab. Oversampling will repeat the histogram generation for the specified number of times and return the average.

The *MonteCarlo* method operates on a continuous parameter space and does not suffer from the problems associated with the gridding for the *Histogram Library* method. It can be slower than the other methods, depending on the desired accuracy as determined by the oversampling factor.

Maximum Likelihood estimator

The *MLE* method does not rely on the simulation of a histogram and thus completely avoids the necessity to bin the data. Instead, it computes the likelihood that the model parameters may produce the data directly. As such, no live plot update is possible. The fit result is finally displayed by means of simulation by the *MonteCarlo* method.

The *MLE* method, as the *MonteCarlo* method, operates on a continuous parameter space without any gridding. It generally shows a smoother optimization surface than the other two methods, and is thus suited well to be used with gradient-based fit algorithms.

9.1.9 Dynamic-PDA

Dynamic interconversion between different FRET states leads to characteristic mixing of FRET efficiencies dependent on the rates of interconversion. This usually shows through the formation of a bridge between the FRET states in the histogram.

Dynamic-PDA, as described in [Kalinin2010a], can be activated by clicking the *Dynamic Model* checkbox in the *Settings* tab. This will enable a two-state kinetic model using the first two species of the fit model. The respective amplitudes in the fit parameter table will be changed to forward and backward rates k_{12} and k_{21} between the two states in unit of ms^{-1}.

Currently, only the two-state model is implemented. Additional static populations can be added.

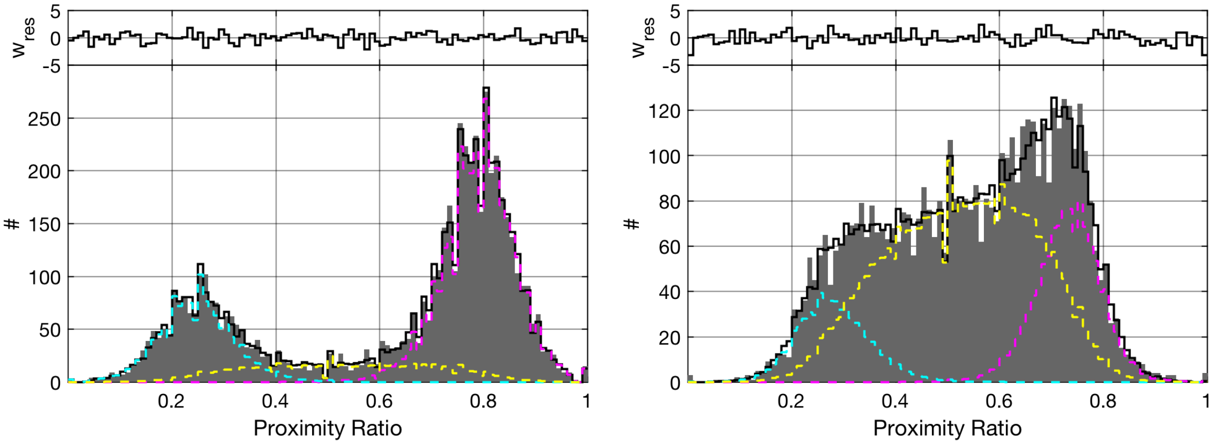


Fig. 5: Example dynamic-PDA analyses of slow (> ms) and fast (< ms) interconversion rates.

Note: It is recommended to globally fit a series of proximity ratio histograms generated at different time bins from the same measurement to obtain a higher accuracy in the determined rates.

9.1.10 Error estimation

PDAFit offers to options to estimate the confidence intervals of the fit parameters, available through the menu *Fit -> Estimate Error*.

Estimate Error from Jacobian at solution: This option will use the Jacobian at the fit result to estimate the confidence intervals of the parameters.

Estimate Error from Markov-chain Monte Carlo: This will perform a Markov-Chain Monte Carlo chain over the χ^2-surface to estimate the confidence intervals of the parameters.

The output will be returned into the current workspace.

9.1.11 Background deconvolution

PDA can be used to describe a complete single-molecule FRET measurement without burst selection. In this case, many of the recorded time bins will only contain background signal and the resulting photon count distribution will be vastly different from the true distribution of fluorescence signal. See the section in the **PAM** manual for how to *export a total measurement to PDA*.

PDAFit implements the deconvolution method as proposed in [Kalinin2008] to recover the distribution of fluorescence signal from the observed photon count distribution using the know distribution of background counts.

This option only applies to the *Histogram Library* method.

Note: Only use this option when the total measurement is analyzed. This method does not apply for burst-selected data sets!

9.2 Program structure

9.2.1 File Menu

Load Files(s): Removes all current data and starts a new list. The program will query multiple files until the user selects *Cancel*.

Add File(s): Adds files to the current list.

Save to File(s): Everything that was opened is saved back into the *.pda* file, including the results from the fit table.

Export Figure(s): Exports figures of the fit results to a subfolder. A *.tif* file of the All tab is saved, a *.tif* of each single file is also saved.

Reload parameters: Re-loads the original parameters table from the loaded files.

Reload Fit parameters: Re-loads the default or saved fit parameters from the loaded files.

9.2.2 Database Tab

Files that are loaded or added via the file menu are put into a list in the database tab. Files can be deleted from this list using the keyboard.

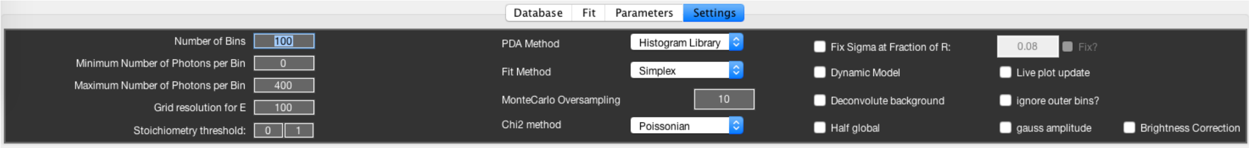
Save: The list can be saved in a PDA Database File (*.pab*), so it can be opened at a later point in time. Importantly, the database file just contains a reference to the filename and path of all files in it.

Load: Load a previously saved database list (*.pab*).

9.2.3 Fit Table

To remove a component from the fit model fix the amplitude *A* to 0. Unchecking a file's 'Active' checkbox removes that file from analysis, but not from the database. Clicking an 'All' row cell value applies this value to all rows.

9.2.4 Settings Tab



Number of Bins: 100
Minimum Number of Photons per Bin: 0
Maximum Number of Photons per Bin: 400
Grid resolution for E: 100
Stoichiometry threshold: 0 1
PDA Method: Histogram Library
Fit Method: Simplex
MonteCarlo Oversampling: 10
Chi2 method: Poissonian
□ Fix Sigma at Fraction of R: 0.08 □ Fix?
□ Dynamic Model
□ Live plot update
□ Deconvolute background
□ ignore outer bins?
□ Half global
□ gauss amplitude
□ Brightness Correction

The *Settings* tab allows you to set a number of parameters that affect the analysis and visualization. The options are explained in more detail in the respective sections. Here, an overview and brief description is given.

Number of Bins: Specifies the number of bins used to construct the proximity ratio histogram over the interval [0,1]. This will effect the display of the histogram, and may effect the fit result of the *Histogram Library* and *MonteCarlo* methods. The *MLE* method is unaffected by this setting.

Minimum/maximum Number of Photons per Bin: Restrict the selection of time bins (see *the respective section*).

Grid resolution for E: Defines the grid resolution over the apparent FRET efficiency for the *Histogram Library* method. The method will construct a library of shot-noise limited proximity ratio histograms at the grid values (e.g. at setting 100, at $\epsilon = [0, 0.01, 0.02, \ldots, 0.98, 0.99, 1]$). It is not advised to set this value lower than 100. Increasing this value with increase the accuracy of the analysis at higher computational cost. This setting has no effect on the PDA methods *MonteCarlo* or *MLE*.

Stoichiometry threshold: Set threshold to restrict the selection of time bins based on the stoichiometry parameter (see *the respective section*).

PDA Method: Set the method for the PDA analysis between *Histogram Library*, *MLE* and *MonteCarlo*. See *the respective section* for details.

Fit Method: Set the fit method to optimize the parameters. Choose between *Simplex* (*fminsearch* function), *Gradient-based* (using the *fmincon* or *lsqnonlin* function), *Patternsearch* or *Gradient-based (global)*. See the MATLAB documentation for more explanation on these fit methods. Generally, the *Simplex* method yields good results at acceptable convergence time.

MonteCarlo Oversampling: This parameter determines the oversampling when using the *MonteCarlo* method. Increasing this number will reduce the statistical noise in the simulated histogram at the cost of increased computation time. The other methods are not affected by this parameter.

Chi2 method: Choose if the χ^2 goodness-of-fit estimator should be calculated based on *Poissonian* or *Gaussian* counting statistics.

Fix Sigma at fraction of R: Check this to fix the distribution width σ at a fixed fraction of the mean distance of a species, i.e. $\sigma = k * R$ where k is typically in the range between 0.05 to 0.2. Set the proportionality factor k using the edit box. The proportionality factor will by default be optimized by the fitting routine. You can additionally fix the proportionality factor to a custom value by checking the *Fix?* checkbox.

Dynamic Model: Check this checkbox to enable dynamic-PDA. Only works in conjunction with the *Histogram Library* method. See *the respective section* for details.

Deconvolute background: Performs deconvolution of the photon count distribution to obtain the background-free fluorescence photon count distribution. Use this only when fitting a whole measurement data set without burst selection. See *the respective section* for more details.

Half global: Allow to set parameters global between subsets of the loaded measurements. Please inquire about details on how to use this functionality.

Live plot update: Enabling this option will cause a live plot update during the fitting process. Note that this slows down the fitting since the plotting takes a significant amount of time. Use this option to check the status of the fitting routine, but generally leave it off.

Ignore outer bins: This options ignores the outermost bins for the calculation of the χ^2 goodness-of-fit estimator, i.e. the bins [0,1/N] and [1-1/N,1] where N is the number of bins in the proximity ratio histogram. This option can be useful if large deviations between model and data are observed for these bins.

gauss amplitude: Enabling this checkbox will scale distance distribution fit result plot with the respective amplitudes of the species. Disable this if you mainly want to investigate changes in the distance and not in the respective population sizes.

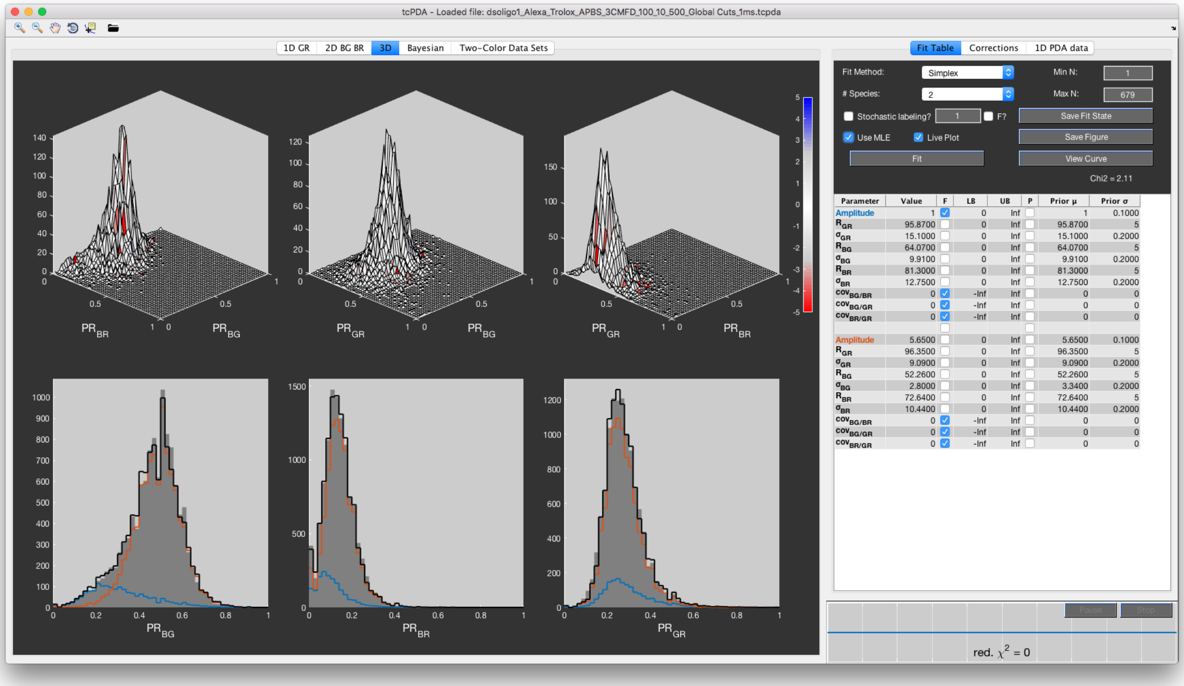
Brightness correction: Corrects for brightness differences between different species using a donor-only brightness reference. This feature is untested and experimental.

CHAPTER 10

tcPDA

Contents

tcPDA (three-color photon distribution analysis) is the module for quantitative analysis of three-color FRET data to extract the underlying distance distributions, similar to the *PDAFit* module for two-color FRET data. It implements the algorithms described in [Barth2018].



Fit Method: Simplex

Species: 2

Min N: 1

Max N: 679

□ Stochastic labeling? 1 □ F?

☑ Use MLE ☑ Live Plot

Fit | Save Fit State | Save Figure | View Curve

Chi2 = 2.11

Parameter	Value	F	LB	UB	P	Prior μ	Prior σ
Amplitude	1	☑	0	Inf	□	1	0.1000
R_{GR}	95.8700	□	0	Inf	□	95.8700	5
σ_{GR}	15.1000	□	0	Inf	□	15.1000	0.2000
R_{BG}	64.0700	□	0	Inf	□	64.0700	5
σ_{BG}	9.9100	□	0	Inf	□	9.9100	0.2000
R_{BR}	81.3000	□	0	Inf	□	81.3000	5
σ_{BR}	12.7500	□	0	Inf	□	12.7500	0.2000
$cov_{BG/BR}$	0	☑	-Inf	Inf	□	0	0
$cov_{BG/GR}$	0	☑	-Inf	Inf	□	0	0
$cov_{BR/GR}$	0	☑	-Inf	Inf	□	0	0
		□			□		
Amplitude	5.6500	□	0	Inf	□	5.6500	0.1000
R_{GR}	96.3500	□	0	Inf	□	96.3500	5
σ_{GR}	9.0900	□	0	Inf	□	9.0900	0.2000
R_{BG}	52.2600	□	0	Inf	□	52.2600	5
σ_{BG}	2.8000	□	0	Inf	□	3.3400	0.2000
R_{BR}	72.6400	□	0	Inf	□	72.6400	5
σ_{BR}	10.4400	□	0	Inf	□	10.4400	0.2000
$cov_{BG/BR}$	0	☑	-Inf	Inf	□	0	0
$cov_{BG/GR}$	0	☑	-Inf	Inf	□	0	0
$cov_{BR/GR}$	0	☑	-Inf	Inf	□	0	0

Fig. 1: The main GUI of tcPDA

10.1 Data loading

10.1.1 From BurstBrowser

To process and export burst data from the *BurstBrowser* module to *tcPDA*, use the *PDA export functionality* in *BurstBrowser*. Load data (*.tcpda* files) by clicking on the folder icon in the top left.

10.1.2 From text file

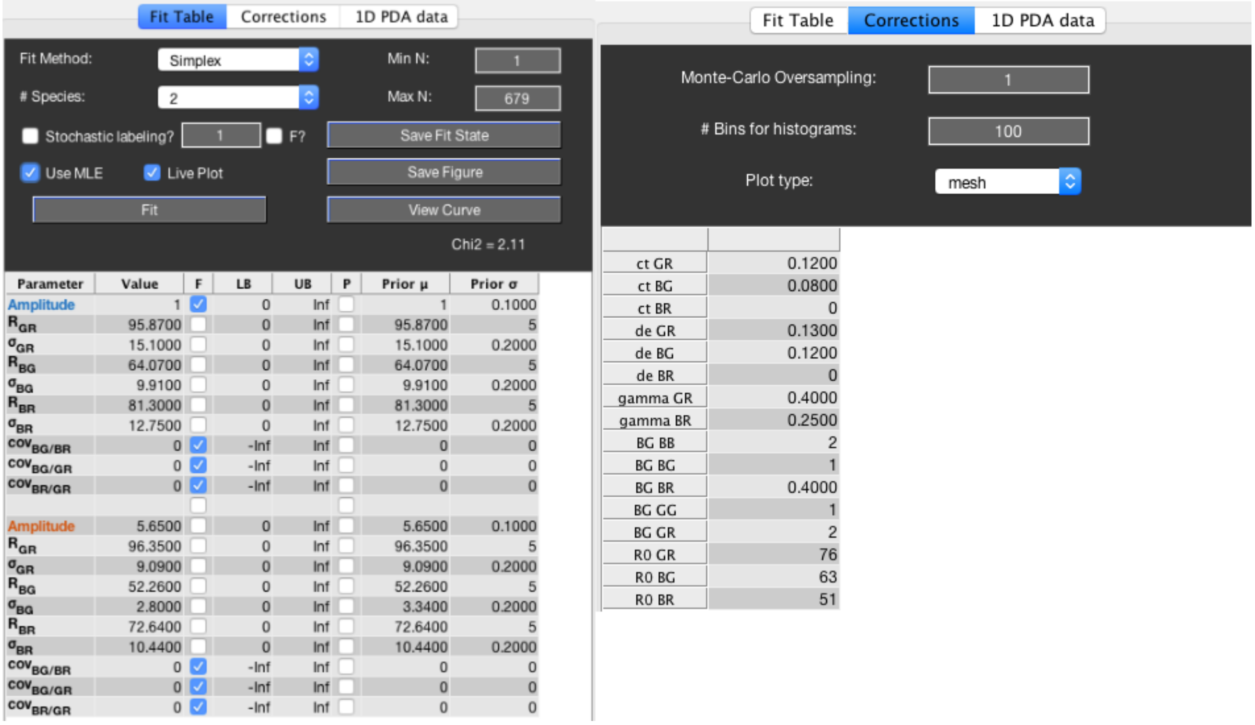
As an alternative to the MATLAB-based *.tcpda* file format, it is also possible to load text-based data into the *tcPDA* program. The text-based file format is a comma-separated file containing the time bin size in milliseconds and a list of the photon counts per time bin for all detection channels $N_{BB}, N_{BG}, N_{BR}, N_{GG}, N_{GR}$.

The basic structure of the text-based file format with ending *.txt* is given by the following template.

```
# place for comments
timebin = 1; # timebin in ms
NBB, NBG, NBR, NGG, NGR
46,54,16,95,28
35,60,17,113,46
38,58,14,82,33
67,51,10,94,23
20,32,6,36,13
32,35,9,46,23
7,37,20,50,14
...
...
```

As before, load it using the folder icon in the top left, and select "Text-based tcPDA file" in the file dialog.

10.2 Fitting data



Fit Method: Simplex

Species: 2

Min N: 1

Max N: 679

☐ Stochastic labeling? 1 ☐ F?

☑ Use MLE ☑ Live Plot

Chi2 = 2.11

Parameter	Value	F	LB	UB	P	Prior μ	Prior σ
Amplitude	1	☑	0	Inf	☐	1	0.1000
R_{GR}	95.8700	☐	0	Inf	☐	95.8700	5
σ_{GR}	15.1000	☐	0	Inf	☐	15.1000	0.2000
R_{BG}	64.0700	☐	0	Inf	☐	64.0700	5
σ_{BG}	9.9100	☐	0	Inf	☐	9.9100	0.2000
R_{BR}	81.3000	☐	0	Inf	☐	81.3000	5
σ_{BR}	12.7500	☐	0	Inf	☐	12.7500	0.2000
$cov_{BG/BR}$	0	☑	-Inf	Inf	☐	0	0
$cov_{BG/GR}$	0	☑	-Inf	Inf	☐	0	0
$cov_{BR/GR}$	0	☑	-Inf	Inf	☐	0	0
		☐			☐		
Amplitude	5.6500	☐	0	Inf	☐	5.6500	0.1000
R_{GR}	96.3500	☐	0	Inf	☐	96.3500	5
σ_{GR}	9.0900	☐	0	Inf	☐	9.0900	0.2000
R_{BG}	52.2600	☐	0	Inf	☐	52.2600	5
σ_{BG}	2.8000	☐	0	Inf	☐	3.3400	0.2000
R_{BR}	72.6400	☐	0	Inf	☐	72.6400	5
σ_{BR}	10.4400	☐	0	Inf	☐	10.4400	0.2000
$cov_{BG/BR}$	0	☑	-Inf	Inf	☐	0	0
$cov_{BG/GR}$	0	☑	-Inf	Inf	☐	0	0
$cov_{BR/GR}$	0	☑	-Inf	Inf	☐	0	0

Monte-Carlo Oversampling: 1

Bins for histograms: 100

Plot type: mesh

ct GR	0.1200
ct BG	0.0800
ct BR	0
de GR	0.1300
de BG	0.1200
de BR	0
gamma GR	0.4000
gamma BR	0.2500
BG BB	2
BG BG	1
BG BR	0.4000
BG GG	1
BG GR	2
R0 GR	76
R0 BG	63
R0 BR	51

Fig. 2: Left: The Fit Table tab contains options for the fit routine and model functions. Right: The Corrections tab allows to set the correction factors.

10.2.1 Defining the correction factors

The correction factors can be set in the *Corrections* tab. If data has been exported from the *BurstBrowser* module, the values for crosstalk (*ct*), gamma-factors (*gamma*), Förster radii (*R0*) and background counts in kHz (*BG*) will be automatically transferred to *tcPDA* and need not be set. However, the value for the **direct excitation probabilities** (*de*) is different in *tcPDA* compared to the intensity-based corrections to photon counts in *BurstBrowser* (see also the respective section in the manual for *PDAFit*). The direct excitation probability of dye X after excitation by laser Y, where $X, Y \in \{B, G, R\}$, is denoted by p_{de}^{XY} (called *de* in the GUI). It is defined based on the extinction coefficients $\varepsilon_X^{\lambda_Y}$:

$$p_{de}^{XY} = \frac{\varepsilon_X^{\lambda_Y}}{\sum_{i=B,G,R} \varepsilon_i^{\lambda_Y}}$$

The other parameters in the *Corrections* tab are explained in the further sections.

10.2.2 The fit parameter table

The fit parameter table allows to set the initial values for the parameters. It adapts to the selected number of species, whereby the color of the *Amplitude* parameter indicates the color of the species in the plots. Check the *F* option to fix parameters. Lower and upper boundaries (*LB* and *UB*) are by default set to 0 and infinity, but can be specified if needed. The other columns are related to the usage of prior information and are described in the *respective section*.

Note: The amplitudes are normalized by the fit routine, i.e. $\sum_i A_i = 1$. To reduce the number of free fit parameters, the amplitude of the first species is fixed by default, but can be unfixed by the user.

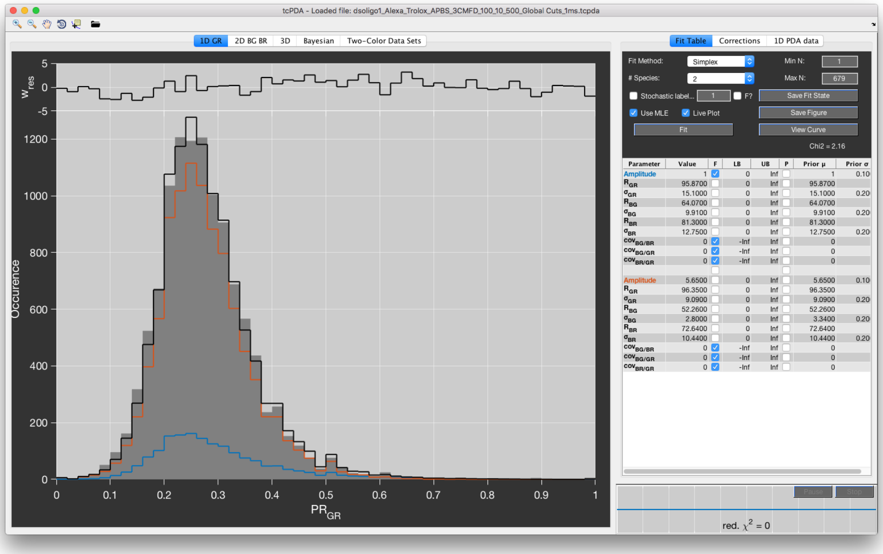
10.2.3 Performing the fit

Start the fit by clicking the *Fit* button in the *Fit Table* tab. The fit options are described in detail *here*. To simply view the theoretical histograms of the current model, click the *View Curve* button.

It is usually best to perform the fit of the three-color FRET histogram in three steps of increasing complexity. Since alternating excitation of the blue, green and red lasers is employed, the direct excitation of the green dye is essentially equivalent to performing a two-color FRET experiment, which can be fit first (*1D GR* tab). Then, one can fit the distribution of the proximity ratios BG and BR while keeping the center value and standard deviation for the distance R_{GR} fixed (*2D BG BR* tab). Finally, the full three-dimensional distribution of the proximity ratio GR, BG and BR can be fit by setting all parameters free (*3D* tab).

The following sections describe the individual steps. The *Fit Table* tab applies to all three tabs (*1D GR*, *2D BG BR* and *3D*) in the main window and adapts functionality based on which main window is selected.

Fitting the histogram of proximity ratio GR



tcPDA - Loaded file: dsoligo1_Alexa_Trolox_APBS_3CMFD_100_10_500_Global Cuts_1ms.tcpda

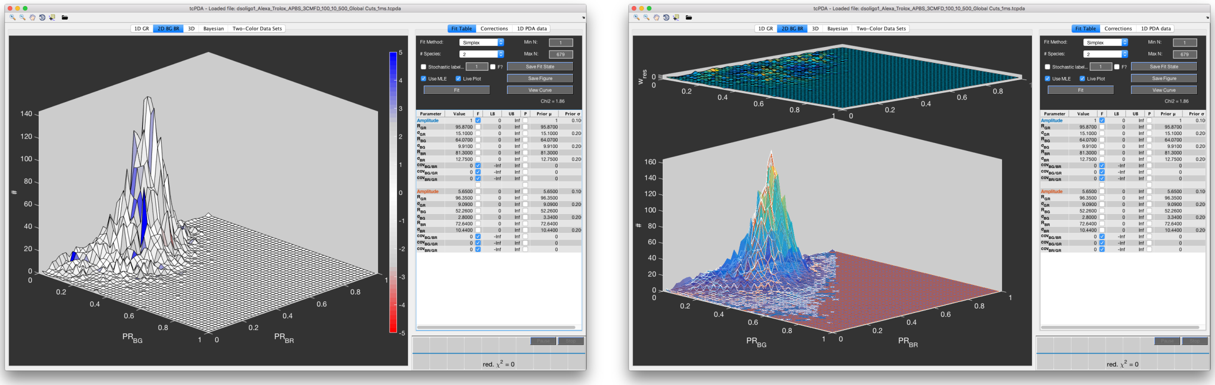
Fit Method: Simplex
Species: 2
Min N: 1
Max N: 679
□ Stochastic label... 1 □ F?
☑ Use MLE ☑ Live Plot
Chi2 = 2.16

Parameter	Value	F	LB	UB	P	Prior μ	Prior σ
Amplitude	1	☑	0	Inf	□	1	0.10
R_{GR}	95.8700	□	0	Inf	□	95.8700	
σ_{GR}	15.1000	□	0	Inf	□	15.1000	0.20
R_{BG}	64.0700	□	0	Inf	□	64.0700	
σ_{BG}	9.9100	□	0	Inf	□	9.9100	0.20
R_{BR}	81.3000	□	0	Inf	□	81.3000	
σ_{BR}	12.7500	□	0	Inf	□	12.7500	0.20
$cov_{BG/BR}$	0	☑	-Inf	Inf	□	0	
$cov_{BG/GR}$	0	☑	-Inf	Inf	□	0	
$cov_{BR/GR}$	0	☑	-Inf	Inf	□	0	
		□			□		
Amplitude	5.6500	□	0	Inf	□	5.6500	0.10
R_{GR}	96.3500	□	0	Inf	□	96.3500	
σ_{GR}	9.0900	□	0	Inf	□	9.0900	0.20
R_{BG}	52.2600	□	0	Inf	□	52.2600	
σ_{BG}	2.8000	□	0	Inf	□	3.3400	0.20
R_{BR}	72.6400	□	0	Inf	□	72.6400	
σ_{BR}	10.4400	□	0	Inf	□	10.4400	0.20
$cov_{BG/BR}$	0	☑	-Inf	Inf	□	0	
$cov_{BG/GR}$	0	☑	-Inf	Inf	□	0	
$cov_{BR/GR}$	0	☑	-Inf	Inf	□	0	

Fig. 3: Fitting the one-dimensional histogram of the proximity ratio GR.

The *1D GR* tab allows fitting of the one-dimensional distribution of the proximity ratio GR, disregarding the available three-color information. It is useful to pre-fit the values for R_{GR} and σ_{GR} and get a first idea of the number of populations required to describe the data. Only the amplitudes and parameters R_{GR} and σ_{GR} are fit. Fitting of the one-dimensional histogram of proximity ratio GR is always performed using Monte Carlo simulation of the histogram with the Simplex fit algorithm, regardless of the fit options set in the *Fit Table* tab. The settings *Use MLE*, *Stochastic Labeling* and *Fit Method* thus have no effect.

Fitting the two-dimensional histogram of proximity ratios BG and BR



tcPDA - Loaded file: dsoligo1_Alexa_Trolox_APBS_3CMFD_100_10_500_Global Cuts_1ms.tcpda

Fit Method: Simplex
Species: 2
Min N: 1
Max N: 679
□ Stochastic label... 1 □ F?
☑ Use MLE ☑ Live Plot
Chi2 = 1.86

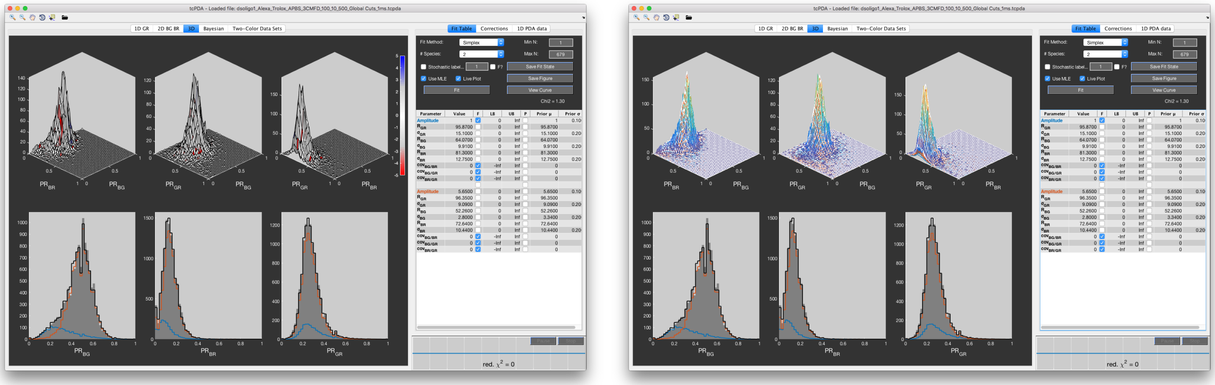
Parameter	Value	F	LB	UB	P	Prior μ	Prior σ
Amplitude	1	☑	0	Inf	□	1	0.10
R_{GR}	95.8700	□	0	Inf	□	95.8700	
σ_{GR}	15.1000	□	0	Inf	□	15.1000	0.20
R_{BG}	64.0700	□	0	Inf	□	64.0700	
σ_{BG}	9.9100	□	0	Inf	□	9.9100	0.20
R_{BR}	81.3000	□	0	Inf	□	81.3000	
σ_{BR}	12.7500	□	0	Inf	□	12.7500	0.20
$cov_{BG/BR}$	0	☑	-Inf	Inf	□	0	
$cov_{BG/GR}$	0	☑	-Inf	Inf	□	0	
$cov_{BR/GR}$	0	☑	-Inf	Inf	□	0	
		□			□		
Amplitude	5.6500	□	0	Inf	□	5.6500	0.10
R_{GR}	96.3500	□	0	Inf	□	96.3500	
σ_{GR}	9.0900	□	0	Inf	□	9.0900	0.20
R_{BG}	52.2600	□	0	Inf	□	52.2600	
σ_{BG}	2.8000	□	0	Inf	□	3.3400	0.20
R_{BR}	72.6400	□	0	Inf	□	72.6400	
σ_{BR}	10.4400	□	0	Inf	□	10.4400	0.20
$cov_{BG/BR}$	0	☑	-Inf	Inf	□	0	
$cov_{BG/GR}$	0	☑	-Inf	Inf	□	0	
$cov_{BR/GR}$	0	☑	-Inf	Inf	□	0	

Fig. 4: Fitting the two-dimensional histogram of the proximity ratios BG and BR. Left: Colormap-representation of the weighted residuals. Right: Mesh-plot of the fit result.

Similarly to the *1D GR* tab, the *2D BG BR* tab is used to fit the two-dimensional distribution of proximity ratios BG and BR. This distribution is sensitive to all three center distances and associated distribution widths, including, albeit indirectly, R_{GR} and σ_{GR}. Leaving R_{GR} and σ_{GR} as free fit parameters, however, will usually lead to values inconsistent with the related distribution of proximity ratio GR, since this information is excluded from this fit. It is thus required to fix R_{GR} and σ_{GR}, which can be easily done by *right-clicking* the fit parameter table and selecting *Fix GR Parameters*. The amplitudes as estimated from the one-dimensional fit may be left free or fixed in this step. Covariances should be fixed in this step and only be refined after center distances and distribution widths have been reasonably fit.

As for the *1D GR* tab, the fit is always performed using Monte Carlo simulation of the histogram with the Simplex fit algorithm, regardless of the fit options set in the *Fit Table* tab.

Fitting the full three-dimensional histogram of photon counts



Fit Method: Simplex
# Species: 2
Min N: 1
Max N: 679
☐ Stochastic label... 1 ☐ F?
☑ Use MLE ☑ Live Plot
Chi2 = 1.30

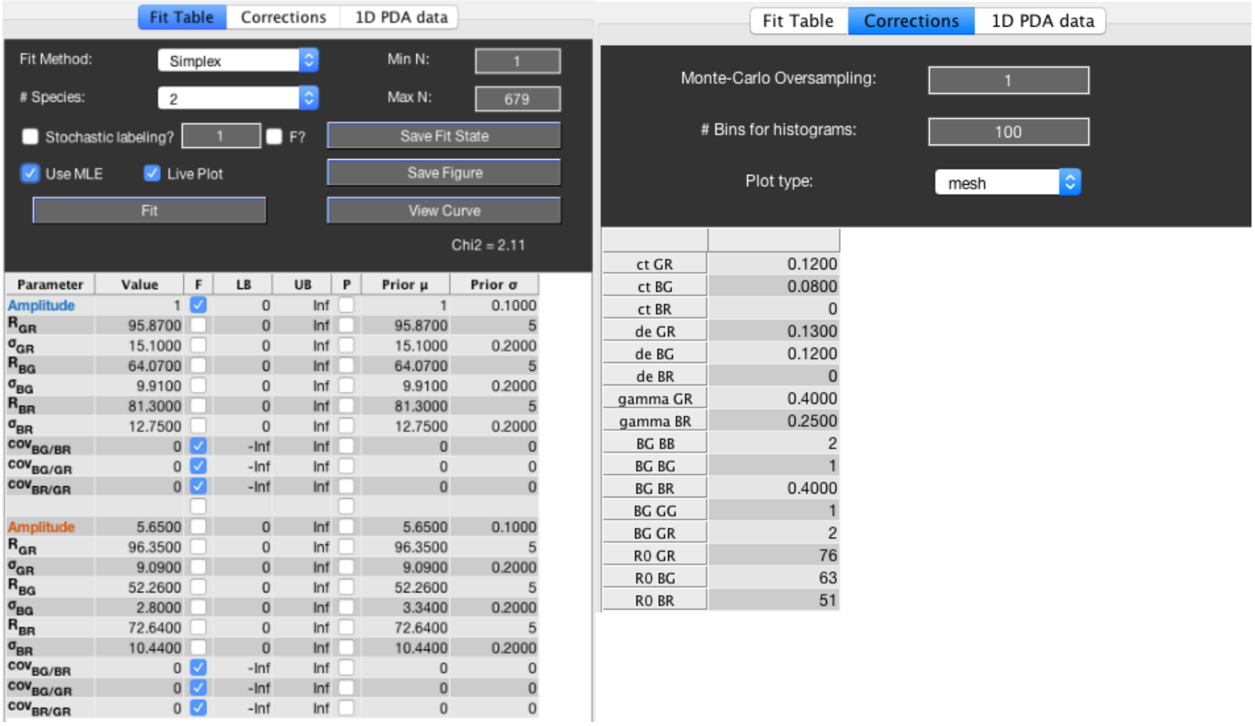
Parameter	Value	F	LB	UB	P	Prior μ	Prior σ
Amplitude	1	☑	0	Inf	☐	1	0.10
R_{GR}	95.8700	☐	0	Inf	☐	95.8700	
σ_{GR}	15.1000	☐	0	Inf	☐	15.1000	0.20
R_{BG}	64.0700	☐	0	Inf	☐	64.0700	
σ_{BG}	9.9100	☐	0	Inf	☐	9.9100	0.20
R_{BR}	81.3000	☐	0	Inf	☐	81.3000	
σ_{BR}	12.7500	☐	0	Inf	☐	12.7500	0.20
$cov_{BG/BR}$	0	☑	-Inf	Inf	☐	0	
$cov_{BG/GR}$	0	☑	-Inf	Inf	☐	0	
$cov_{BR/GR}$	0	☑	-Inf	Inf	☐	0	
		☐			☐		
Amplitude	5.6500	☐	0	Inf	☐	5.6500	0.10
R_{GR}	96.3500	☐	0	Inf	☐	96.3500	
σ_{GR}	9.0900	☐	0	Inf	☐	9.0900	0.20
R_{BG}	52.2600	☐	0	Inf	☐	52.2600	
σ_{BG}	2.8000	☐	0	Inf	☐	3.3400	0.20
R_{BR}	72.6400	☐	0	Inf	☐	72.6400	
σ_{BR}	10.4400	☐	0	Inf	☐	10.4400	0.20
$cov_{BG/BR}$	0	☑	-Inf	Inf	☐	0	
$cov_{BG/GR}$	0	☑	-Inf	Inf	☐	0	
$cov_{BR/GR}$	0	☑	-Inf	Inf	☐	0	

Fig. 5: Fitting the full three-dimensional histogram of the proximity ratios GR, BG and BR. Left: Colormap-representation of the weighted residuals. Right: Mesh-plot of the fit result.

The determined parameters should now reasonably describe the data and the refinement of the fit can be performed first using Monte Carlo simulation of histograms. This step is necessary since the maximum likelihood estimator (MLE) will return zero likelihood when the parameter values are too far from their optimum value. Once a region of high likelihood has been reached, one can switch from the Monte Carlo estimation to the MLE. To check if an area of high posterior density has been reached, enable the *Use MLE* checkbox and confirm that the reported log-likelihood *logL* is not -Inf.

For the fitting of the full three-dimensional histogram of photon counts, it is advised to iteratively fix and un-fix parameters. Start e.g. by leaving only the center distances as free parameters, after which the distribution width can be fit. Shortcuts for the fixing and un-fixing of groups of parameters are available by *right-clicking* the fit parameter table. As a final step, un-fix the covariances. Using the Simplex algorithm, covariances will usually not diverge from their initial zero values. Instead, try using the gradient-based or pattern search fit algorithms to fit the covariances. Finally, all parameters should be left free and refined simultaneously. This step can be time consuming, but is essential to ensure that the obtained result is the correct minimum.

10.2.4 Options and settings



Fit Method: Simplex

Min N: 1

Species: 2

Max N: 679

□ Stochastic labeling? 1 □ F?

☑ Use MLE ☑ Live Plot

Chi2 = 2.11

Parameter	Value	F	LB	UB	P	Prior μ	Prior σ
Amplitude	1	☑	0	Inf	□	1	0.1000
R_{GR}	95.8700	□	0	Inf	□	95.8700	5
σ_{GR}	15.1000	□	0	Inf	□	15.1000	0.2000
R_{BG}	64.0700	□	0	Inf	□	64.0700	5
σ_{BG}	9.9100	□	0	Inf	□	9.9100	0.2000
R_{BR}	81.3000	□	0	Inf	□	81.3000	5
σ_{BR}	12.7500	□	0	Inf	□	12.7500	0.2000
$cov_{BG/BR}$	0	☑	-Inf	Inf	□	0	0
$cov_{BG/GR}$	0	☑	-Inf	Inf	□	0	0
$cov_{BR/GR}$	0	☑	-Inf	Inf	□	0	0
		□			□		
Amplitude	5.6500	□	0	Inf	□	5.6500	0.1000
R_{GR}	96.3500	□	0	Inf	□	96.3500	5
σ_{GR}	9.0900	□	0	Inf	□	9.0900	0.2000
R_{BG}	52.2600	□	0	Inf	□	52.2600	5
σ_{BG}	2.8000	□	0	Inf	□	3.3400	0.2000
R_{BR}	72.6400	□	0	Inf	□	72.6400	5
σ_{BR}	10.4400	□	0	Inf	□	10.4400	0.2000
$cov_{BG/BR}$	0	☑	-Inf	Inf	□	0	0
$cov_{BG/GR}$	0	☑	-Inf	Inf	□	0	0
$cov_{BR/GR}$	0	☑	-Inf	Inf	□	0	0

Monte-Carlo Oversampling: 1

Bins for histograms: 100

Plot type: mesh

ct GR	0.1200
ct BG	0.0800
ct BR	0
de GR	0.1300
de BG	0.1200
de BR	0
gamma GR	0.4000
gamma BR	0.2500
BG BB	2
BG BG	1
BG BR	0.4000
BG GG	1
BG GR	2
R0 GR	76
R0 BG	63
R0 BR	51

Fig. 6: Left: Options related to the fit procedure. Right: Display of the photon count distribution found in the Corrections tab.

Fit options

A number of options are available regarding the fit routine and model function.

To switch between using the likelihood expression or $\chi^2_{red.}$ estimator, use the **Use MLE** checkbox. If checked, the likelihood expression is evaluated and used as the objective function of the fit routine. If unchecked, the $\chi^2_{red.}$ estimator is used based on Monte Carlo simulation of the shot-noise limited histograms. In this case, the *Monte Carlo Oversampling* parameter in the *Corrections* tab determines the used oversampling factor.

The *Fit Method* dropdown menu allows to choose between different optimization algorithms. If the *1D GR* or *2D BG BR* tabs are selected, the *Simplex* method is always used, regardless of the choice in the *Fit Method* dropdown menu.

The following fit methods can be chosen:

Simplex: Based on the fminsearch function of MATLAB. To allow to set lower and upper parameter bounds, the fminsearchbnd function from the *MATLAB File Exchange* is used instead.

Pattern Search: Based on the patternsearch function of the *Global Optimization Toolbox* of MATLAB. This function is less likely to be trapped in a local minimum than the *Simplex* option.

Gradient-based: Based on the fmincon function of MATLAB. Uses gradient-based optimization algorithm and is thus generally faster to converge than the *Simplex* algorithm, but poses a higher risk to end in a local minimum.

Gradient-based (global): Based on the GlobalSearch algorithm of the *Global Optimization Toolbox* of MATLAB. Similar to the *Gradient-based* fit algorithm, but performs multiple optimization runs using the *fmincon* routine using randomized starting points.

The options related to the model function are:

# Species: Specify the number of species of the model function.

Stochastic labeling: Toggle the use of the stochastic labeling correction. This option includes a second population for every species with permuted distances R_{BG} and R_{BR}. Use this option if not all fluorophores are attached site-specifically. The current implementation assumes that the blue fluorophore is attached site-specifically, but that the two acceptor fluorophores are stochastically distributed over the two possible label positions. Specify the value for the fraction *F* of the species matching the distance assignment in the fit table using the edit box. The permuted species then has contribution *(1-F)*. To fix the stochastic labeling fraction, toggle the *F?* checkbox. If unchecked, the stochastic labeling fraction is optimized by the fit routine as a free parameter, taking the specified value as the initial value.

Options that affect the computation time are:

Min N: Exclude time bins with less photons than the specified value. Inspect the photon count distribution shown at the bottom of the *Corrections* tab to judge the effect of this selection.

Max N: Exclude time bins with more photons than the specified value. Inspect the photon count distribution shown at the bottom of the *Corrections* tab to judge the effect of this selection.

Monte Carlo Oversampling: Increase this value to reduce statistical noise in the simulated histograms due to the random number generation. The Monte Carlo sampling step will be performed *N* times, and the final histogram is divided by *N*.

Display options

Live Plot: If checked, the plots and fit parameter table will update at every fit iteration. If *Use MLE* is checked, only the fit parameter table is updated since the theoretical histogram is not calculated during the fit process. The *Live Plot* option will slow down the fit routine due to the time it takes to update the plots.

Plot type: This option is found in the *Corrections* tab. Specify the way that two-dimensional histograms are plotted. The *Mesh* option will plot the data as a surface and show the fit function as a mesh. For the *2D BG BR* tab. the weighted residuals are plotted above (see *here*). The *Colormap* option will plot only the data in the two-dimensional plots. Weighted residuals are shown using a colormap on the data in a range between -5 and 5, color-coded from red over white to blue. See *here* to see the effect of this option on the *3D* tab.

# Bins for histogram: Specify the number of bins per dimension used to histogram the data. Since a higher number of bins will result in less counts per bin overall, one should also increase the Monte Carlo Oversampling* parameter when increasing the number of bins to counteract the increased noise due to the reduced sampling.

Saving the fit state

Saving the current settings and fit parameters using the *Save Fit State* button will store all information in the *.tcpda* file. To save different fit states for the same file, right-click the *Save Fit State* button and select *Save Fit State in separate file* to store the parameters in a *.fitstate* file. To load a fit state from an external file, right-click the *Save Fit State* button and select *Load Fit State from separate file*. Loaded two-color PDA data sets are also saved upon saving the fit state.

Exporting figures

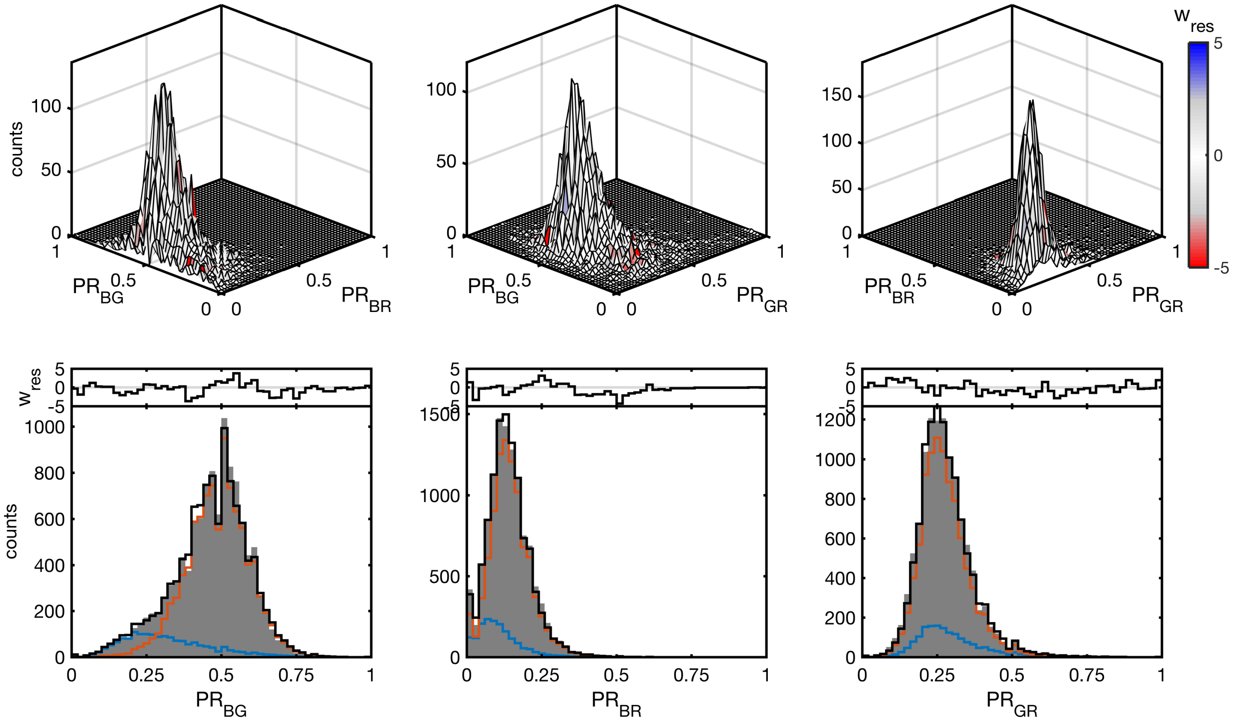


Fig. 7: Exported figure of the fit result.

To export the fit result to a figure, click the *Save Figure* button. This will open a new figure window and plot the currently displayed photon count distribution using the *colormap* display option. Additionally, a second plot will be generated showing the extracted distance distributions and correlation coefficients between the three distance dimensions.

Note: Before exporting a figure, re-generate the simulated histogram using a high value for the Monte Carlo oversampling parameter to reduce stochastic noise.

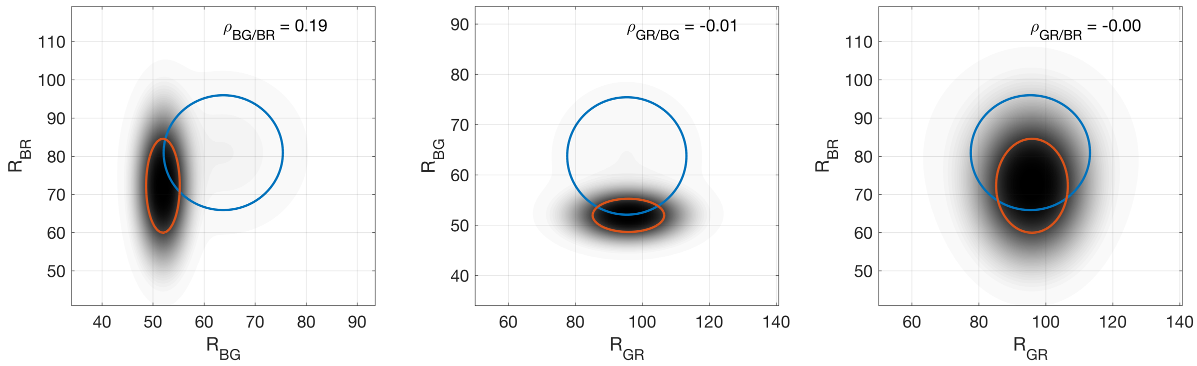


Fig. 8: Marginal distributions of the fitted distance distribution. Correlation coeffcients between the distance pairs are given in the top right of the plots. Colored circles indicate 1σ levels of the individual species.

10.2.5 Using prior information

In the Bayesian framework, prior information about the parameters can be included into the analysis by the prior distribution:

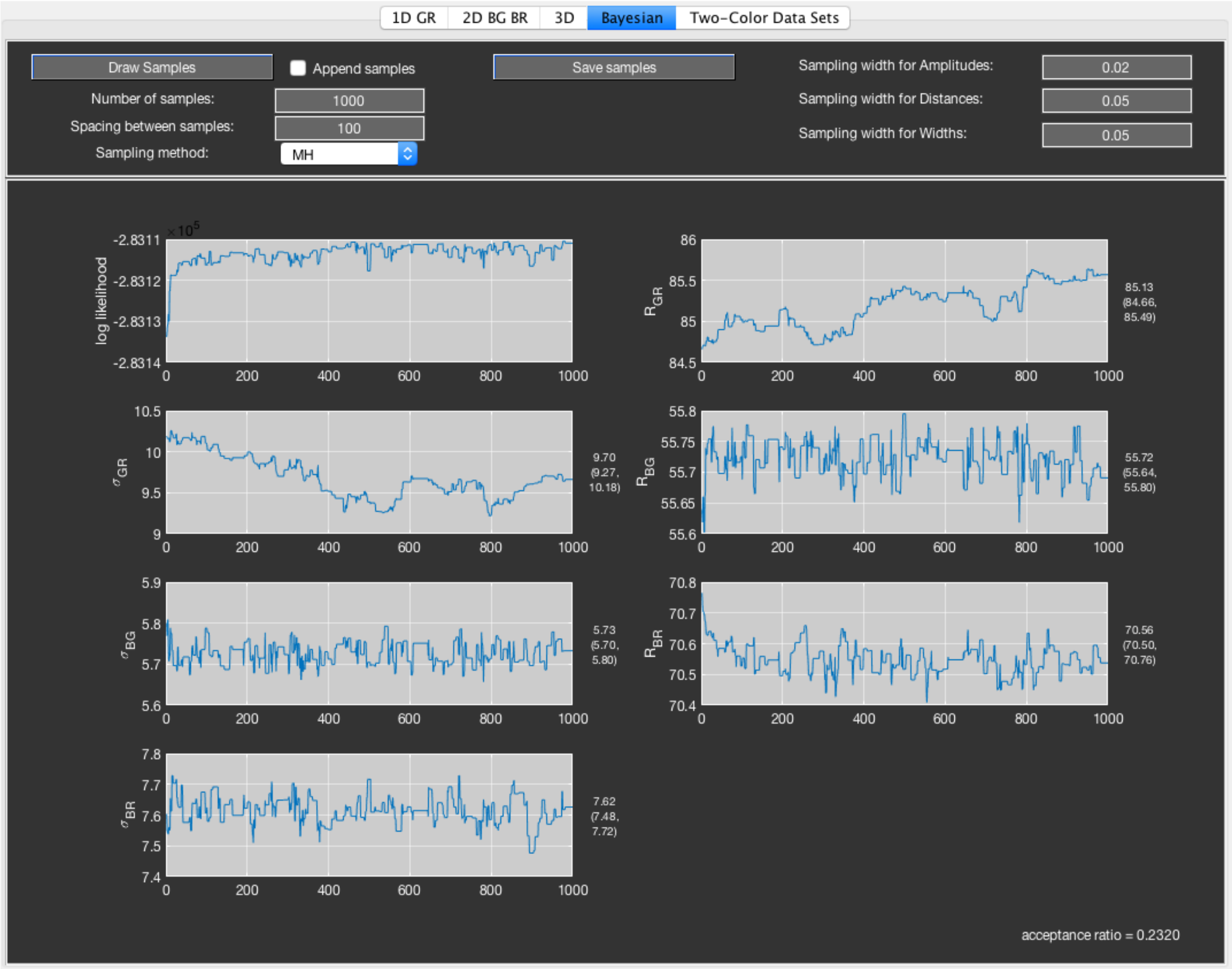
$$P(\theta|D,I) \propto P(D|\theta)P(\theta|I)$$

Here, $P(\theta|D,I)$ is the posterior probability of the parameter vector θ given the data D and the background information I, which is proportional to the probability of the data given the parameters, $P(D|\theta)$, multiplied by the prior distribution of the parameters given the background information, $P(\theta|I)$.

In *tcPDA*, one can assign normal prior distribution for each parameter separately using the *fit parameter table*. Checking the *P* checkbox for a parameter enables the prior for it. Specify the mean (Prior μ) and width (Prior σ) for the parameter as estimated from the available information.

Only normally distributed prior are available, and no correlated information about different parameters is accounted for at the moment.

10.2.6 Estimating confidence intervals



☐ Append samples

Number of samples: 1000

Spacing between samples: 100

Sampling method: MH

Sampling width for Amplitudes: 0.02

Sampling width for Distances: 0.05

Sampling width for Widths: 0.05

acceptance ratio = 0.2320

Fig. 9: Markov chain Monte Carlo sampling of the posterior probability density.

The *Bayesian* tab allows to estimate confidence intervals of the fitted parameters using Markov chain Monte Carlo (MCMC) sampling. A random walk is performed over the parameter space. At every step, new parameter values are drawn based on the proposal probability distribution given by a Normal distribution with specified width. The change of the parameter vector is accepted with probability given by the ratio of the likelihoods:

$$p = \min\left(\frac{\mathcal{L}_{n+1}}{\mathcal{L}_\backslash}, 1\right)$$

By tuning the width of the proposal distribution, one can adjust the acceptance probability of the random walk, which should fall into the range of 0.2-0.5.

This method to sample the posterior likelihood is called Metropolis-Hasting sampling (called *MH* in the *Sampling method* dropdown menu). Additionally, *tcPDA* implements Metropolis-within-Gibbs sampling (called *MWG* in the *Sampling method* dropdown menu). This sampling method performs a Metropolis-Hasting sampling step for all parameters consecutively. The simultaneous re-sampling of all parameters in MH sampling can lead to low acceptance probabilities. By performing the sampling one parameter at a time, the algorithm is less likely to step into regions of low likelihood. MWG is thus to be preferred to MH when many parameters are estimated simultaneously. However, since the likelihood has to be evaluated at every step for every parameter, it is computationally more expensive.

Based on the sampled parameter values, the 95% confidence intervals are determined from the standard deviation σ by CI $= 1.96\sigma$. To obtain independent samples from the posterior distribution, one can define a spacing between the samples used to determine the confidence intervals.

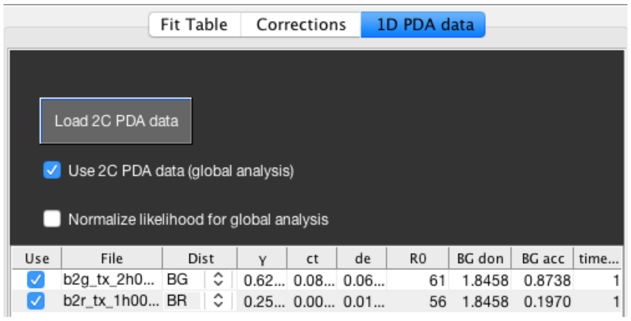
Click the *Draw Samples* button in the *Bayesian* tab to begin drawing the number of samples from the posterior distribution as specified in the *Number of samples* input field. Drawing samples takes the same parameters and options defined in the *Fit table* and *Corrections* tabs as performing a normal fit, with the difference that the MLE is always used. Choose the sampling method between MH and MWG using the *Sampling method* dropdown menu. When the *Append samples* checkbox is selected, the newly drawn samples will be added to the previous run of the random walk. Otherwise, previous results will be overwritten. The display will update every 100 steps and show the evolution of the log-likelihood and all free parameters. The acceptance ratio is shown in the bottom right and periodically updated. The sampling width can be set separately for the amplitudes, center distances and distribution width, and can be adjusted to tune the acceptance ratio. Upon completion of the sampling, mean parameter values and 95% confidence intervals will be shown to the right of the parameter plots. Confidence intervals are determined from samples with specified spacing in the *Spacing between samples* field. This value can be adjusted after the sampling has completed, which will update the displayed confidence intervals. Finally, click the *Save samples* button to store the result of the MCMC sampling in a text file in the folder of the data.

10.2.7 Global analysis using two-color data sets

tcPDA also implements a global analysis of two- and three-color FRET data sets. This function is useful if the three-color system has previously been studied using two-color FRET experiments with the same labeling positions. Hereby, the fit functions optimizes a global likelihood over all loaded data sets:

$$\log \mathcal{L}_{\}\updownarrow\wr\lfloor\dashv\updownarrow} = \log \mathcal{L}_{\ni\mathcal{C}} + \sum_i \log \mathcal{L}_{\in\mathcal{C},\rangle}$$

The underlying distances are the same between the two- and three-color FRET data sets. However, the correction factors and size of the time bin can be different between the experiments. It is even possible to include two-color FRET information measured using different dyes, assuming that the fluorophores are not influencing the properties of the studied molecule.



☑ Use 2C PDA data (global analysis)

□ Normalize likelihood for global analysis

Use	File	Dist	γ	ct	de	R0	BG don	BG acc	time...
☑	b2g_tx_2h0...	BG	0.62...	0.08...	0.06...	61	1.8458	0.8738	1
☑	b2r_tx_1h00...	BR	0.25...	0.00...	0.01...	56	1.8458	0.1970	1

Fig. 10: GUI for handling additional two-color PDA data.

Currently, only two-color PDA data exported from the *BurstBrowser* module is supported (*.pda* files). The correction factors and time bin are read from the *.pda* file, but can be modified in the table as well. Every two-color data set needs to be assigned the corresponding distance in the three-color FRET system (*GR*, *BG* or *BR*). Toggle the use of a specific data set by unchecking the *Use* checkbox in the table. Loaded two-color data sets are stored when the *fit state* is saved. The two-color FRET data sets are shown in the *Two-color Data Sets* tab.

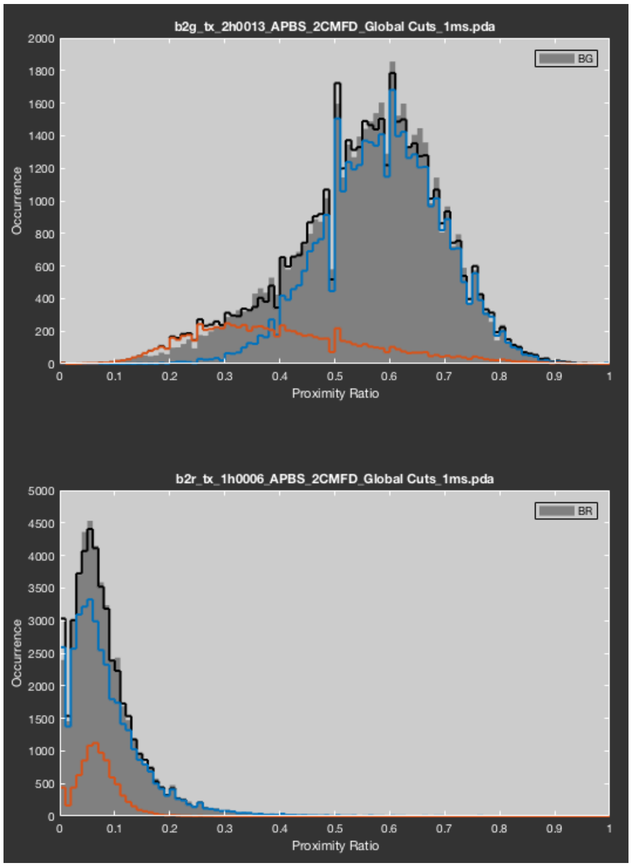


Fig. 11: Display of the fit results for additional two-color FRET data sets.

Check the *Use 2C PDA data (global analysis)* to include the two-color information in the fit. The *Normalize likelihood for global analysis* checkbox will use the geometric mean of the individual likelihoods to assign equal weight to each data set:

$$\log \sqrt[n]{L} = \frac{\log \mathcal{L}}{n}$$

Consider that using many two-color FRET data sets with this option will reduce the overall contribution of the three-color data set.

Note: Note that, while information on the center distances, population sizes and distribution widths is shared between two- and three-color data sets, the correlated information is unique to the three-color data set. Thus, the use of the global analysis is expected to increase the robustness of the extracted correlated information.

10.3 Considerations for fit performance

10.3.1 Monte Carlo simulation of histograms

The number of bins for the histograms affects this fitting procedure. The Monte Carlo oversampling parameter, found in the *Corrections GUI*, will reduce noise and thus improve the convergence of the fitting procedure, however the computation time will scale linearly with this parameter. For initial fitting, a value of 1 may be chosen to quickly find the optimal region. Upon refinement, the value should be increased to 10 or higher.

The simulated histograms are used to represent the fit result also when the MLE procedure is used. To export a plot of the final histograms, showing the result of the analysis, consider using a high oversampling factor of 100 to obtain theoretical histograms that are not affected by noise.

10.3.2 Maximum likelihood estimator

The MLE is unaffected by the number of bins used for this histograms or the Monte Carlo oversampling parameter. However, the performance of the likelihood calculation critically depends on the background count numbers. Try to keep the background signal low to improve performance of the algorithm.

The performances decrease in presence of high background count rates because the total likelihood is given by the sum over all likelihoods given the possible combinations of background and fluorescence photon counts. Since three channels are present in three-color FRET, the number of likelihood evaluations is thus given by the product of the number of background counts to consider per channel, which quickly grows to significant excess calculations.

Consider for example a background count rate of 1 kHz per channel and a time bin size of 1 ms, corresponding to an average background count number of 1. By accounting for the probabilities of 0, 1 and 2 background counts, given by a Poissonian distribution, one accounts for more than 90% of the possible background count occurrences. Still, the trinomial distribution would have to be evaluated $3 * 3 * 3 = 27$-times more often (and the binomial distribution 9-times more often) compared to when no background signal were present.

CUDA support

The likelihood estimator is available for use with NVIDIA GPUs. By default, the program will look for available CUDA-compatible graphics cards and default to the CPU if none are found.

For use in MATLAB, the CUDA code is compiled into a MATLAB MEX file and available for the Windows and Linux operating systems. If the supplied MEX file does not work, follow the instructions in the help file found with the source code to re-compile the MEX file on your computer.

CHAPTER 11

Particle Detection for Phasor

This module is designed to detect particles or other regions (e.g. cells) and pool all pixels for the individual regions for better signal-to-noise. As this module is currently in a very early development stage, its shape and functions are changing rapidly. Therefore, it will be explained only very briefly at this moment.

The program requires framewise phasor data from **PAM** (*.phf* files) that are loaded and displayed.

At this moment, a threshold and a wavelet based detection algorithms are implemented.

This algorithms use the intensity information of the file itself or alternatively an external mask (e.g. a second color channel), based on TIFF data, to detect the regions.

The data is then saved as a single, summed up image with an averaged phasor for all pixels of an individual region. The phasors of the pixels not assigned can be either saved normally or set to zero.

Alternatively, the data can be saved as a trace for each particle. Hereby, an image is created, where the x-axis corresponds to the frame and the y-axis to the particle number.

Both types of data are saves as *.phr* files to be analysed the the phasor module.

CHAPTER 12

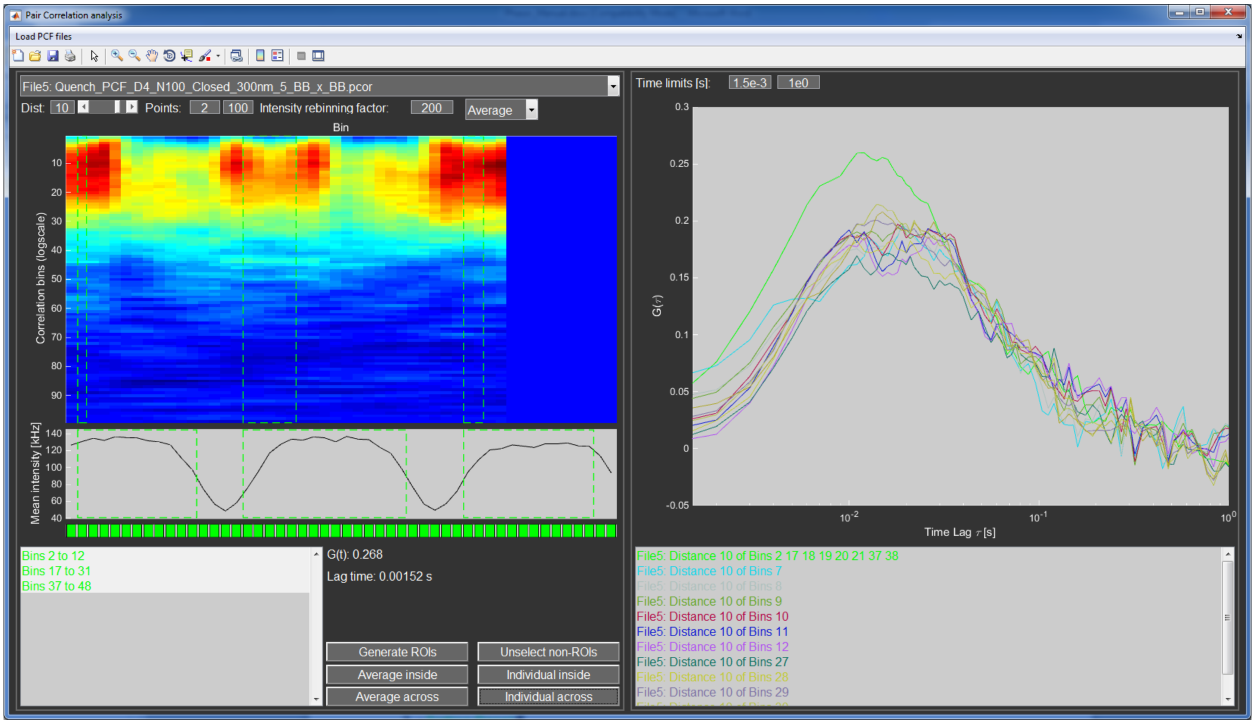
Pair Correlation Analysis

Contents

The pair correlation analysis window can be opened from the PAM main window by clicking on **Advanced Analysis -> PCF Analysis** or from the command window by typing in `PCFAnalysis`.

The pair correlation method ([Digman2009]) extracts information about the spatial behavior of molecular movement using rapid line-scanning and a correlation function between pairs of spatial locations in the sample. It is especially useful to measure (spatially and directionally resolved) diffusion and transport of molecules across barriers.

The **PCF Analysis** window is divided into two main parts. On the left side the PCF carpet is displayed and the user can select regions and extract individual correlation curves. The right side is reserved for display of the individual correlation curves that can be averaged and exported.



File5: Quench_PCF_D4_N100_Closed_300nm_5_BB_x_BB.pcor

Dist: 10 Points: 2 100 Intensity rebinning factor: 200 Average

Time limits [s]: 1.5e-3 1e0

Bins 2 to 12
Bins 17 to 31
Bins 37 to 48

G(t): 0.268
Lag time: 0.00152 s

File5: Distance 10 of Bins 2 17 18 19 20 21 37 38
File5: Distance 10 of Bins 7
File5: Distance 10 of Bins 8
File5: Distance 10 of Bins 9
File5: Distance 10 of Bins 10
File5: Distance 10 of Bins 11
File5: Distance 10 of Bins 12
File5: Distance 10 of Bins 27
File5: Distance 10 of Bins 28
File5: Distance 10 of Bins 29

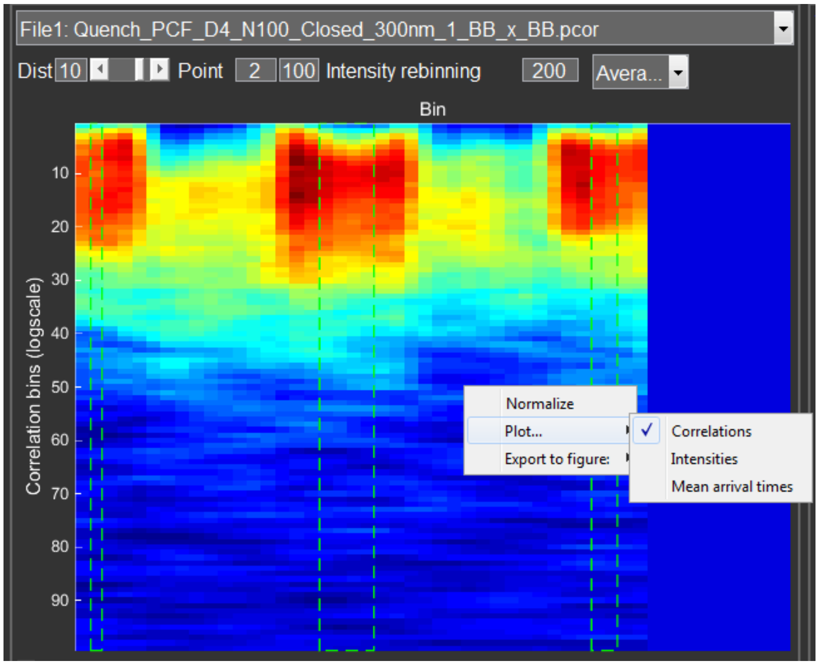
Fig. 1: The main GUI of PCF analysis.

12.1 Data handling and loading data

Currently, the program only supports files created with **PAM** with the file ending *.pcor*. These files are based on standard *.mat* files and can also be accessed in the MATLAB command window.

To load new files, got to **Load PCF files** and select either **Load New File** or **Add Files**. **Load New File** removes previously loaded files, while **Add Files** adds the new one to the old ones. The currently selected file can be selected with a popupmenu. The loaded data are stored in the global variable *PCFData*.

12.2 PCF Carpet display



File1: Quench_PCF_D4_N100_Closed_300nm_1_BB_x_BB.pcor

Dist 10 Point 2 100 Intensity rebinning 200 Avera...

Fig. 2: Displaying the pair correlation carpet.

Pair correlation data is a 4 dimensional matrix (position, time lag, distance lag, and backwards and forwards correlation). This makes it very hard to properly display the data in a meaningful way. In this program a carpet for a single distance lag is displayed, with the laser position (=spatial bin) as the horizontal axis and the time lags (as a quasi-logarithmic scale) as the vertical axis. The correlation amplitude is colorcoded using the *Jet* colormap.

The user can select the displayed spatial lag with the **Dist** editbox and slider. Furthermore, the editboxes for **Points** restrict the temporal lags displayed, useful to remove artifacts at early or late time points. Initially, the average of the forward and backwards correlation is plotted (to increase S/N). In cases where diffusion might be direction dependent, the user can select to only display and analyze the forward or backward correlations with a popupmenu.

Right clicking the carpet gives the user additional options:

- **Normalize:** Normalizes the carpet to the maximal value for each spatial bin. Selecting it again replots the original data.
- **Plot:** This option allows the user to switch between the correlation, an intensity or an arrival time carpet.
- **Export:** This exports the currently plotted carpet to a new MATLAB figure.

Below the carpet is a plot displaying the mean intensity profile of the scanned region. With a *right click* the user can switch to the mean arrival time profile.

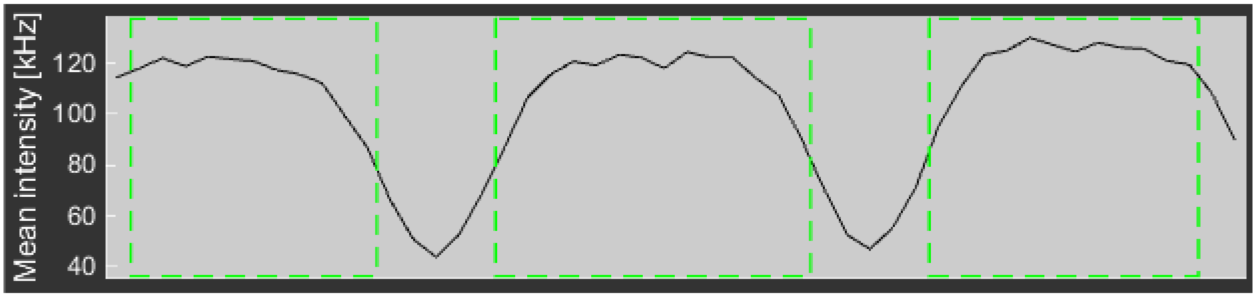


Fig. 3: Intensity profile.

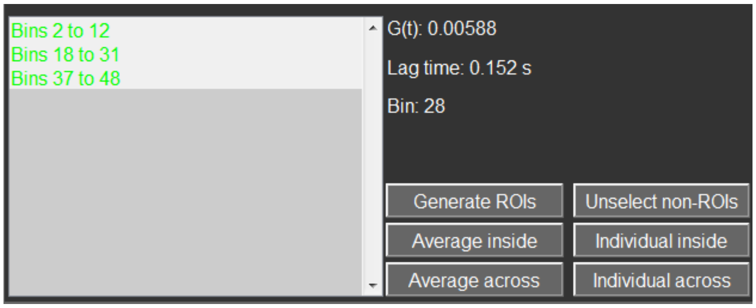
12.3 Selection of bins and ROIs

In most cases not all bins of the carpet are of particular interest for further analysis (e.g. parts outside of a scanned cell) or the bins need to be sorted into different categories.

Individual bins can be unselected with the green boxes directly below the intensity profile plot. A right click unselects a single bin (indicated by a red box). Note that each spatial bin is part of two correlation curves and both are omitted. A second click selects it again. The left mouse button toggles the selection for all bins at once. All further analysis and averaging require the definition of regions of interest (ROIs). ROIs can be generated in several different ways: * **Generate ROIs Button:** Clicking this button will select a ROI for each consecutive stretch of active bins. That means that if all bins are active a single ROI spanning the full range will be selected. * **Left Click in the Intensity Profile:** Left clicking and dragging the mouse in the intensity profile plot selects the area as a single ROI (even across inactive bins) * **Left Click in the Carpet:** Left clicking and dragging the mouse in the carpet also selects a ROI. However, due to the spatial lag, this will generate a larger ROI, since it has to include both spatial bins used for the correlation.

The individual ROIs are shown in a listbox at the bottom left. They can be repositioned using the arrow keys and resized with the ´+´ and ´-´ keys. The standard color of a ROI is green. Selecting them and pressing the c key allows the user to select a new color.

12.4 Curve extraction and averaging


Bins 2 to 12
Bins 18 to 31
Bins 37 to 48

G(t): 0.00588

Lag time: 0.152 s

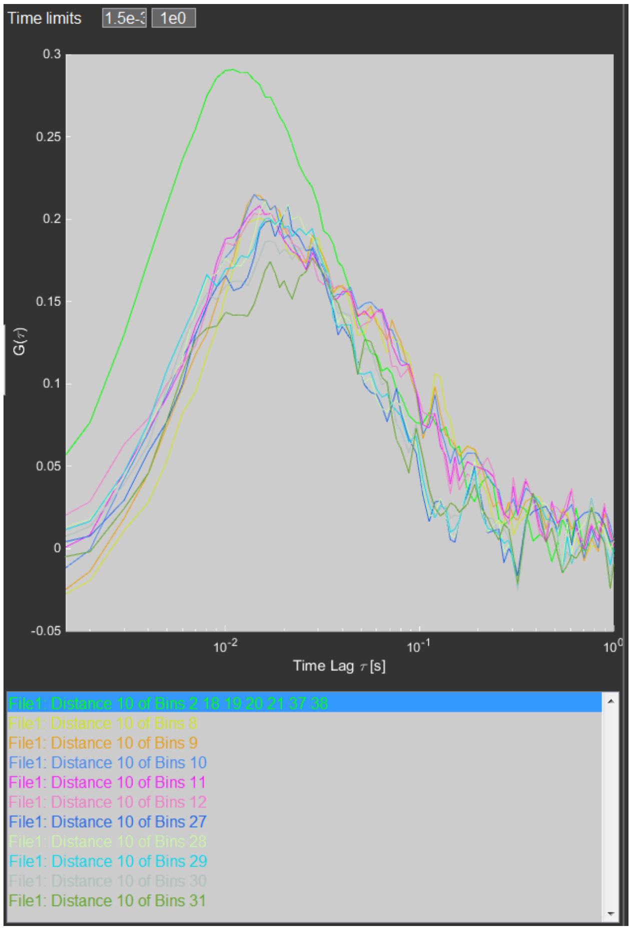
Bin: 28

Fig. 4: Settings for curve extraction and averaging.

Once the ROIs are defined, the user has to extract the relevant bins either as individual or averaged correlation curves to the correlation curve part on the right side of the window. Hereby, the program distinguishes between two different types of curves:

1. Both spatial bins of a particular correlation curve are situated within a single ROI. This case is called “inside? and these curves can be extracted with the **Individual inside** button as an entry for each single one or as a single averaged curve with the **Averaged inside** button or the enter/return key. The individual curves get a random color while the averaged curve has the average color of the ROIs.
2. When the first and second bins of a correlation curve are situated in two different ROIs, the case is called “across?. This requires that at least two ROIs are selected. These curves can be extracted with the **Individual across** button as an entry for each single one or as a single averaged curve with the **Averaged across** button or the enter/return key. The individual curves get a random color while the averaged curve has the inverse of the average color of the ROIs (1-RGB).

12.5 Individual curve display and exporting


Time limits 1.5e-3 1e0

File1: Distance 10 of Bins 2 18 19 20 21 37 38
File1: Distance 10 of Bins 8
File1: Distance 10 of Bins 9
File1: Distance 10 of Bins 10
File1: Distance 10 of Bins 11
File1: Distance 10 of Bins 12
File1: Distance 10 of Bins 27
File1: Distance 10 of Bins 28
File1: Distance 10 of Bins 29
File1: Distance 10 of Bins 30
File1: Distance 10 of Bins 31

Fig. 5: Correlation curve display and exporting.

The individual and averaged curves extracted from the ROIs are displayed and listed on the right side of the program.

The arrow keys can toggle the display of the selected curves. A further averaging of the curves can be achieved by selecting them and pressing the ´+´ key.

Pressing enter/return will export each selected curve as a *.mcor* file that can be loaded and fit with the *FCSFit* program.

Bibliography

[Nir2006] Nir, E. et al. Shot-Noise Limited Single-Molecule FRET Histograms: Comparison between Theory and Experiments. J. Phys. Chem. B 110, (2006).

[Schaffer1999] Schaffer, J. et al. Identification of Single Molecules in Aqueous Solution by Time-Resolved Fluorescence Anisotropy. J. Phys. Chem. A 103 (1999).

[Enderlein1997] Enderlein, J., Robbins, D. L. & Ambrose, W. P. The statistics of single molecule detection: an overview. (1997).

[Tomov2012] Tomov, T. E. et al. Disentangling Subpopulations in Single-Molecule FRET and ALEX Experiments with Photon Distribution Analysis. Biophys J 102 (2012).

[Maus2001] Maus, M. et al. An Experimental Comparison of the Maximum Likelihood Estimation and Nonlinear Least-Squares Fluorescence Lifetime Analysis of Single Molecules. Anal. Chem. 73 (2001).

[Felekyan2005] Felekyan, S. et al. Full correlation from picoseconds to seconds by time-resolved and time-correlated single photon detection. Rev. Sci. Instrum. 76, 083104 (2005).

[Nettels2007] Nettels, D., Gopich, I. V., Hoffmann, A. & Schuler, B. Ultrafast dynamics of protein collapse from single-molecule photon statistics. Proc Natl Acad Sci USA 104 (2007).

[Chen1999] Chen, Y., Mueller, J. D., So, P. T. C. & Gratton, E. The Photon Counting Histogram in Fluorescence Fluctuation Spectroscopy. Biophys J 77 (1999).

[Boehmer2002] Boehmer, M., Wahl, M., Rahn, H. J., Erdmann, R. & Enderlein, J. Time-resolved fluorescence correlation spectroscopy. Chem Phys (2002).

[Felekyan2012] 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, 1036–1053 (2012).

[Otosu2013] Otosu, T., Ishii, K. & Tahara, T. Note: Simple calibration of the counting-rate dependence of the timing shift of single photon avalanche diodes by photon interval analysis. Rev. Sci. Instrum. 84, 036105 (2013).

[Enderlein2005] Enderlein, J. & Gregor, I. Using fluorescence lifetime for discriminating detector afterpulsing in fluorescence-correlation spectroscopy. Rev. Sci. Instrum. 76, 033102 (2005).

[Koshioka1995] Koshioka, M., Sasaki, K. & Masuhara, H. Time-Dependent Fluorescence Depolarization Analysis in Three-Dimensional Microspectroscopy. Applied Spectroscopy 1–5 (1995).

[Lakowicz2007] Lakowicz, J. R. Principles of Fluorescence Spectroscopy. (Springer Science & Business Media, 2007), Chapter 4, Section 11.

[Ivanov2009] Ivanov, V., Li, M. & Mizuuchi, K. Impact of emission anisotropy on fluorescence spectroscopy and FRET distance measurements. Biophys J 97, 922–929 (2009)

[Petersen1993] Petersen, N.O., et al. Quantitation of membrane receptor distributions by image correlation spectroscopy: concept and application. Biophys. J. 65(3), 1135-1146 (1993)

[Wiseman2000] Wiseman, P.W., et al. Two-photon image correlation spectroscopy and image cross-correlation spectroscopy. J. Microsc. 200, 14–25 (2000)

[Digman2005] Digman, M.A., et al. Measuring fast dynamics in solutions and cells with a laser scanning microscope. Biophys. J. 89(2), 1317-1327 (2005)

[Digman2009a] Digman, M.A., et al. Detecting protein complexes in living cells from laser scanning confocal image sequences by the cross correlation raster image spectroscopy method. Biophys. J. 96(2) 96707–716 (2009)

[Toplak2012] Toplak, T., et al. STICCS reveals matrix-dependent adhesion slipping and gripping in migrating cells. Biophys. J. 103(8), 1672–1682 (2012)

[DiRienzo2013] Di Rienzo, C., et al. Fast spatiotemporal correlation spectroscopy to determine protein lateral diffusion laws in live cell membranes. Proc. Natl. Acad. Sci. 110(30) 12307-12312 (2013)

[Digman2008] Digman, M.A., et al. Mapping the number of molecules and brightness in the laser scanning microscope. Biophys. J. 94(6), 2320-2332 (2008)

[Dalal2008] Dalal, R.B., et al. Determination of particle number and brightness using a laser scanning confocal microscope operating in the analog mode. Microsc. Res. Tech. 71(1) 69-81 (2008)

[Digman2009b] Digman, M.A., et al. Stoichiometry of molecular complexes at adhesions in living cells. Proc. Natl. Acad. Sci. 106(7), 2170-2175 (2009)

[Hendrix2013] Hendrix, J., et al. Pulsed interleaved excitation fluctuation imaging. Biophys. J. 105(4) 848-861 (2013)

[Hendrix2015] Hendrix, J., et al. Live-cell observation of cytosolic HIV-1 assembly onset reveals RNA-interacting Gag oligomers. J Cell Biol 210(4) 629-646 (2015)

[Hendrix2016] Hendrix, J., et al. Arbitrary-Region Raster Image Correlation Spectroscopy. Biophys. J 111(8) 1785-1796 (2016)

[Schrimpf2018] Schrimpf, W. et al. Crosstalk-free multicolor RICS using spectral weighting. Methods 140-141:97-111 (2018)

[Longfils2019] Longfils, M. Raster Image Correlation Spectroscopy Performance Evaluation. Biophys. J 117(10):1900-1914 (2019)

[Redfort2005] Redfort, G.I and Clegg, R.M. Polar Plot Representation for Frequency-Domain Analysis of Fluorescence Lifetimes. J Fluoresc 15(5), 805-815 (2005).

[Digman2008] Digman, M.A, et al. The phasor approach to fluorescence lifetime imaging analysis. Biophys. J. 94(2), L14-L16 (2008).

[Schrimpf2016] Schrimpf, W. et al. Investigation of the Co-Dependence of Morphology and Fluorescence Lifetime in a Metal?Organic Framework. Small 12(27), 3651-3657.

[Kudryavtsev2012] Kudryavtsev, V. et al. Combining MFD and PIE for Accurate Single-Pair Förster Resonance Energy Transfer Measurements. ChemPhysChem 13, 1060–1078 (2012).

[Botev2010] Botev, Z. I., Grotowski, J. F. & Kroese, D. P. Kernel density estimation via diffusion. The Annals of Statistics 38, 2916–2957 (2010).

[Lee2005] Lee, N. K. et al. Accurate FRET Measurements within Single Diffusing Biomolecules Using Alternating-Laser Excitation. Biophys J 88, 2939–2953 (2005).

[Kalinin2010] Kalinin, S., Valeri, A., Antonik, M., Felekyan, S. & Seidel, C. A. M. Detection of structural dynamics by FRET: a photon distribution and fluorescence lifetime analysis of systems with multiple states. J. Phys. Chem. B 114, 7983–7995 (2010).

[Laurence2007] Laurence, T. A. et al. Correlation spectroscopy of minor fluorescent species: signal purification and distribution analysis. Biophys J 92, 2184–2198 (2007).

[Kalinin2012] Kalinin, S. et al. A toolkit and benchmark study for FRET-restrained high-precision structural modeling. Nat Meth 9, 1218–1225 (2012)

[Antonik2006] Antonik, M., Felekyan, S., Gaiduk, A. & Seidel, C. A. M. Separating Structural Heterogeneities from Stochastic Variations in Fluorescence Resonance Energy Transfer Distributions via Photon Distribution Analysis. J. Phys. Chem. B 110, 6970–6978 (2006).

[Kalinin2010a] Kalinin, S., Valeri, A., Antonik, M., Felekyan, S. & Seidel, C. A. M. Detection of structural dynamics by FRET: a photon distribution and fluorescence lifetime analysis of systems with multiple states. J. Phys. Chem. B 114, 7983–7995 (2010).

[Kalinin2008] Kalinin, S., Felekyan, S., Valeri, A. & Seidel, C. A. M. Characterizing multiple molecular States in single-molecule multiparameter fluorescence detection by probability distribution analysis. J. Phys. Chem. B 112, 8361–8374 (2008).

[Kalinin2010b] Kalinin, S., Sisamakis, E., Magennis, S. W., Felekyan, S. & Seidel, C. A. M. On the origin of broadening of single-molecule FRET efficiency distributions beyond shot noise limits. J. Phys. Chem. B 114, 6197–6206 (2010).

[Barth2018] Barth, A., Voith von Voithenberg, L., & Lamb, D. C. (2019). Quantitative Single-Molecule Three-Color Förster Resonance Energy Transfer by Photon Distribution Analysis. J Phys Chem B, 123(32), 6901-6916. https://doi.org/10.1021/acs.jpcb.9b02967

[Digman2009] Digman, M.A. Imaging barriers to diffusion by pair correlation functions. Biophys. J. 97(2), 665-673, (2009).

[Lamb2000] Lamb, D. C., Schenk, A., Röcker, C., Scalfi-Happ, C. & Nienhaus, G. U. Sensitivity Enhancement in Fluorescence Correlation Spectroscopy of Multiple Species Using Time-Gated Detection. Biophys J 79, 1129–1138 (2000)

[Müller2005] Müller, B. K., Zaychikov, E., Bräuchle, C. & Lamb, D. C. Pulsed interleaved excitation. Biophys J 89, 3508–3522 (2005)

[Kudryavtsev2012] Kudryavtsev, V. et al. Combining MFD and PIE for Accurate Single-Pair Förster Resonance Energy Transfer Measurements. ChemPhysChem 13, 1060–1078 (2012)

[Hendrix2013] Hendrix, J., Schrimpf, W., Höller, M. & Lamb, D. C. Pulsed Interleaved Excitation Fluctuation Imaging. Biophys J 105, 848–861 (2013)

[Hendrix2016] Hendrix, J., Dekens, T., Schrimpf, W. & Lamb, D. C. Arbitrary-Region Raster Image Correlation Spectroscopy. Biophys J 111, 1785–1796 (2016)