
SMIS Manual

Latest version: 2.1

=====

Copyright (C) 2022 Dominique Bourgeois

E-mail: dominique.bourgeois@ibs.fr

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or any later version. This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

=====

Introduction

- SMIS (Single Molecule Imaging Simulator) is a Matlab-based simulation software that enables to simulate a large variety of single-molecule fluorescence imaging experiments in Widefield mode.
- The novelty of SMIS is the *advanced description of the fluorophores* used in the simulations, notably taking into account their *spectral and photophysical characteristics*.
- Examples of simulations that can be performed include PALM, dSTORM, sptPALM, PAINT, in 2D or 3D.
- Multicolor experiments can be simulated with unlimited number of colors.
- Complex laser excitation schemes can be simulated with unlimited number of lasers.
- Complex diffusion patterns of the fluorophores can be simulated with unlimited number of diffusion states.
- SMIS outputs .tif image stacks and ground truth .mat data.

Reference

Bourgeois, D.; Single Molecule Imaging Simulations with Advanced Fluorophore Photophysics.
<https://www.biorxiv.org/content/10.1101/2022.06.14.496133v2>

Table of content

Introduction.....	1
Reference	1
Table of content	2
Download and Installation	4
Downloading SMIS from Github.....	4
Running SMIS.....	4
SMIS main window	5
SMIS quick help	6
Getting familiar with SMIS:	6
Main steps to design a simulation :.....	6
Defining default directories.....	7
Running simulations	8
Choosing between 2D and 3D simulations.....	8
Choosing the number of fluorophores (or number of colors)	8
Choosing virtual samples.....	8
General characteristics of virtual samples:	8
Loading virtual samples:.....	10
Setting up the fluorophores	12
Defining new fluorophores.....	15
Creating spectra	19
Setting up lasers	21
Lasers sequences.....	23
Setting up Objective and PSF.....	23
Setting up Emission Filters and Acquisition Channels.....	25
Setting up the EMCCD camera	27
Setting up the fluorescence background	28
Setting up sample drift	30
Fluorescence sampling rate.....	31
Other general parameters.....	32
Single Particle Tracking.....	34
sptPALM options	35
Other SMIS features	36
Monitoring what's going on	36
Lasers.....	37

Fluorophore spectra	37
PSF	38
Monitoring data collection	38
Monitoring photophysics	39
Parallel computing	39
Force overwrite	40
Launching simulations	40
Output Data	42
Creating virtual sample	42
Analyzing results	43
Analyzing single molecule data	44
Analyzing ensemble data	48
Annex 1: Content of the <i>sms</i> Matlab structure	49
Annex 2: Content of the <i>sm_par</i> Matlab structure	50
Annex 3: Content of the <i>im_par</i> Matlab structure	51

Download and Installation

The software can be used either as a standalone application for Windows, MacOS or Linux, or as a MATLAB app.



SMIS was developed under Windows. *Proper running of SMIS under MacOS or Linux has not been thoroughly checked.*

Downloading SMIS from Github

- Go to: <https://github.com/DominiqueBourgeois/SMIS>
- For **Windows** and **Linux**:
 - Use the default SMIS branch called *main_LINUX_WINDOWS*
 - Click on the green button “Code”, and select “download ZIP”. This will download the file: “SMIS-main_LINUX_WINDOWS.zip”
- For **MacOS**:
 - Switch to the Github SMIS branch called *MACOS*
 - Click on the green button “Code”, and select “download ZIP”. This will download the file: “SMIS-MACOS.zip”
- Finally, unzip the SMIS .zip file in your preferred directory.

Running SMIS

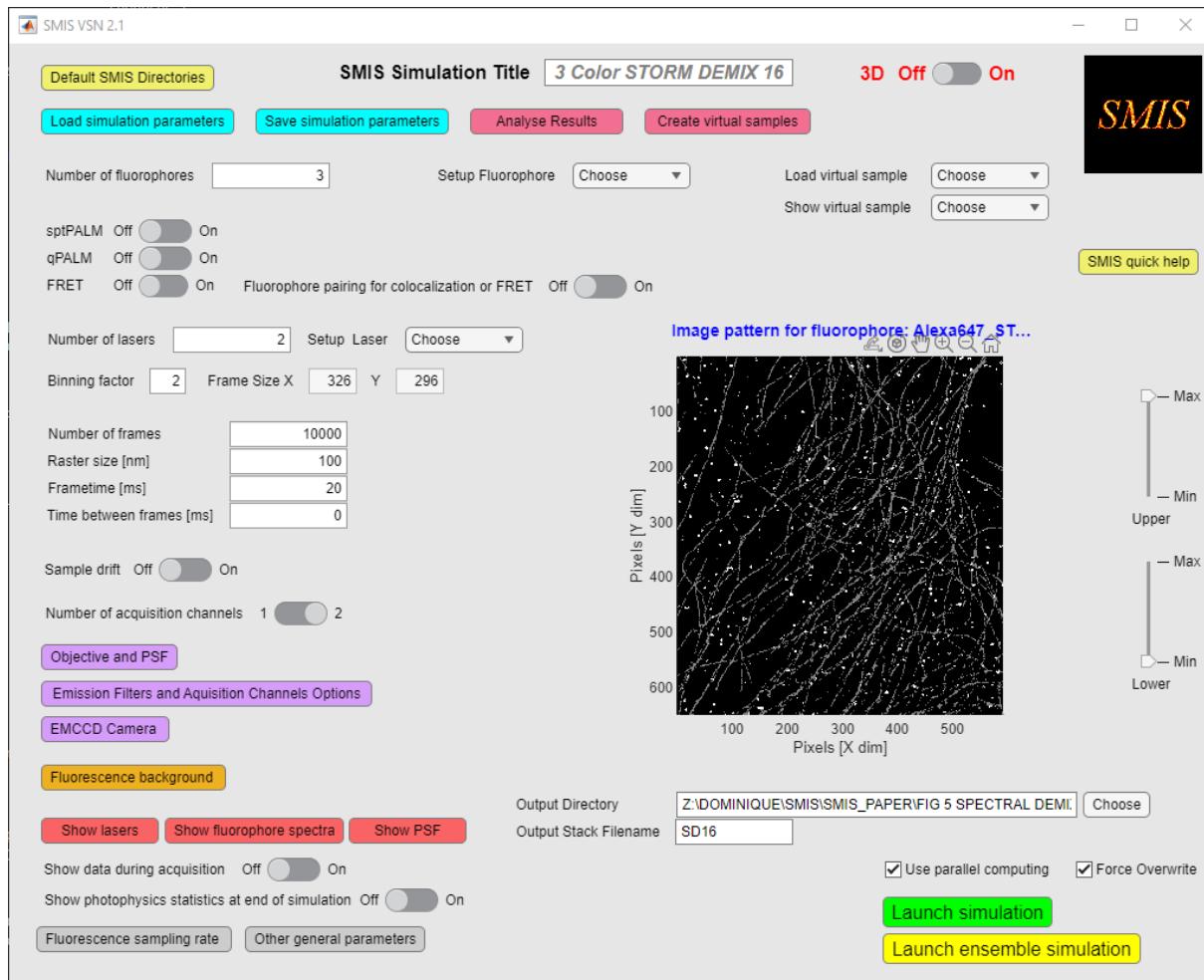
- **Standalone SMIS**
 - To run the standalone SMIS, the Matlab runtime must be installed on your computer. The runtime with proper version should be installed. To download the Matlab runtime, go to <https://fr.mathworks.com/products/compiler/matlab-runtime.html>
 - **Windows**
 - Make sure Matlab runtime 2022a (9.12) is installed on your computer.
 - The SMIS executable is found in: SMIS/STANDALONE/DISTRIBUTE/WINDOWS
 - **To execute SMIS, double-click on “SMIS.exe”**
 - **Linux**
 - Make sure Matlab runtime 2021b (9.11) is installed on your computer.
 - The SMIS executable is found in: SMIS/STANDALONE/DISTRIBUTE/LINUX
 - **To execute SMIS, open a terminal, move to the directory where SMIS is installed and type at the prompt: “./run_SMIS.sh <mcr_directory>”, where <mcr_directory> is the location of the Matlab runtime.**
 - **MacOS**
 - Make sure Matlab runtime 2022b (9.13) is installed on your computer.
 - The SMIS executable is found in: SMIS/STANDALONE/DISTRIBUTE/MACOS

- To execute SMIS, open a terminal, move to the directory where SMIS is installed and type at the prompt: “`./run_SMIS.sh <mcr_directory>`”, where `<mcr_directory>` is the location of the Matlab runtime.

- Using the **SMIS app** with Matlab
 - To properly run SMIS with Matlab, you need the **Image Processing Toolbox**, the **Statistics and Machine Learning Toolbox**, and preferably the **Parallel Computing Toolbox** (not compulsory).
 - In Matlab, go to the APPS tab, and click on “*Install App*”
 - For **Windows** select the “`SMIS_Windows.mlappinstall`” located in SMIS/APP
 - For **Linux** select the “`SMIS_Linux.mlappinstall`” located in SMIS/APP
 - For **MacOS** select the “`SMIS.mlappinstall`” located in SMIS/APP

SMIS main window

Upon starting SMIS, the following main window will open:



SMIS quick help

In SMIS, interactive help is available by moving the mouse to the desired field.

Getting familiar with SMIS:

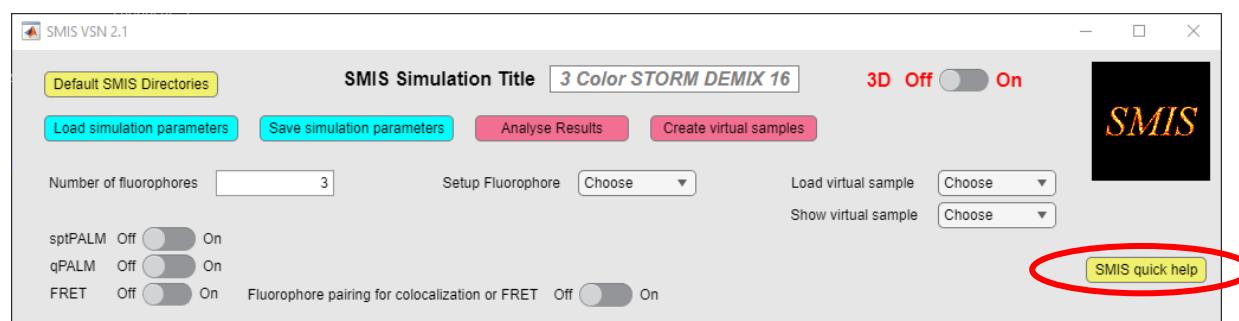
To start learning about SMIS, it is advisable to **load simulation examples** by clicking on: “*Load simulations*”.

Then navigate throughout the different SMIS submenus. Enter a proper output directory for the simulation and click on “Launch simulation”.

Some of the simulations can take very long to run. To run faster, in the “Setup Fluorophore” menu, you may want to reduce the number of used molecules.

Main steps to design a simulation :

The main steps to set up a simulation can be visualized by clicking on the « *SMIS quick help* » button.

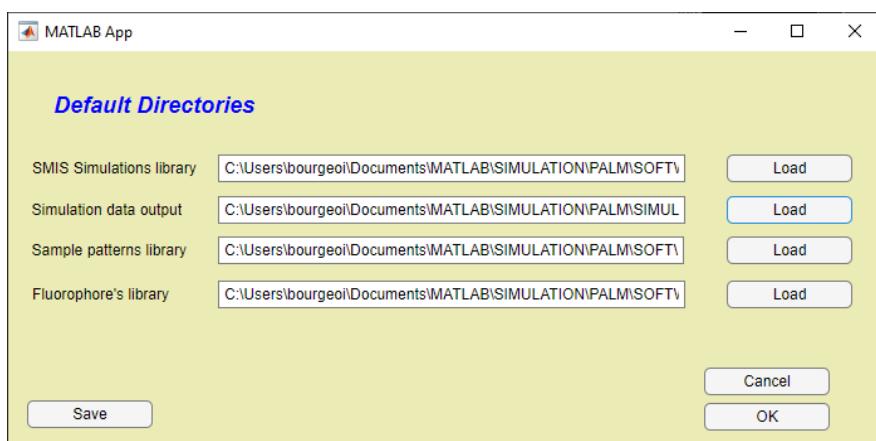


1. If there is a similar simulation available, load it with « *Load simulation parameters* ».
2. Enter the name of your simulation in the « *SMIS Simulation Title* ».
3. Choose whether this is a 2D or 3D simulation with « *3D Off-On* ».
4. Set the number of different fluorophores (e.g. for multicolor experiments) you want to use in « *Number of fluorophores* ».
5. Load the virtual samples for each defined fluorophore via the « *Load virtual sample* » menu. You can create virtual samples with the menu “*Create virtual samples*”.
6. Set up labeling and photophysics for each fluorophore with the « *Setup Fluorophore* » menu. In this menu, you can load existing fluorophores, or create your own fluorophore by entering “*Define new fluorophore*”.
7. Decide whether this is a « *sptPALM* », « *qPALM* » or « *FRET* » experiment. The fluorophores and virtual samples must have been chosen accordingly. You will get warnings if this is not the case.
8. Define the number of lasers to be used and set up each laser with the « *Setup Laser* » menu.
9. Define the final frame size of the detector (and output stack) by setting the « *Binning factor* »

10. Define the « *Number of frames* », « *Raster size* », « *Frametime* » and « *Time between frame time* ».
11. Define eventual « *Sample drift* ».
12. Decide whether this is a single or two-channel experiment with the « *Number of acquisition channels* » toggle.
13. Set up the microscope « *Objective and PSF* » parameters.
14. Set up the « *Emission Filters and Acquisition Channels options*», and eventually the parameters of the « *EMCCD camera* » (such as the EMCCD gain).
15. Define the « *Fluorescence background* ».
16. Choose the « *Output Directory* » and « *Stack File Name* ».
17. Save your simulation with the « *Save simulation parameters* » button. Saving can be repeated at any time during the process.
18. Finally launch the simulation with the « *Launch simulation* » button (single molecule mode) or « *Launch ensemble simulation* » (ensemble mode).
19. The output .tif stacks can be analyzed as real experimental data. The ground truth SMIS data can be analyzed with the « *Analyze Results* » tool.

Defining default directories

Defaults directories can be set up in « *Default SMIS Directories* ».

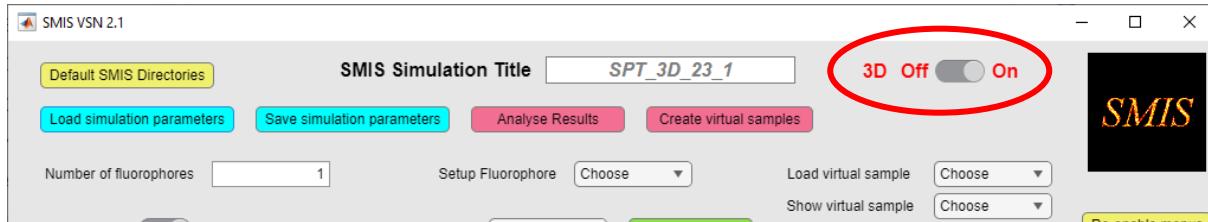


This is useful to directly access your *SMIS simulations library*, *SMIS simulation output directory*, *virtual sample library* and *fluorophore library*.

If you want to save the entered directories for future SMIS sessions, click on « *Save* ».

Running simulations

Choosing between 2D and 3D simulations



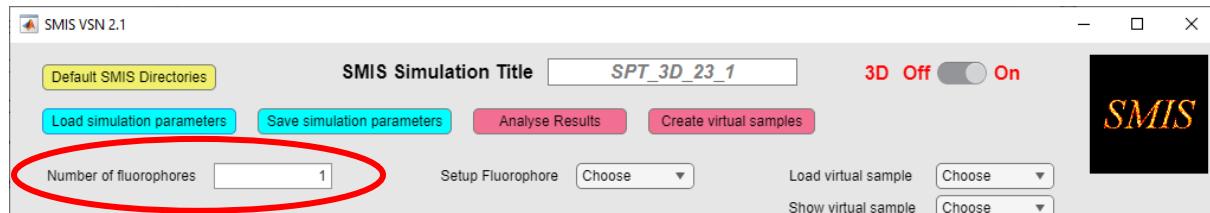
If you want to run a 2D simulation set 3D to « *Off* », and if you want to run a 3D simulation, set 3D to « *On* ».

For 2D simulations, virtual samples are 2D segmented images in a .tif format. Ground truth molecule positions will only be defined in X and Y.

For 3D simulations, virtual samples are 3D Matlab arrays. Ground truth molecule positions will be defined in X, Y and Z.

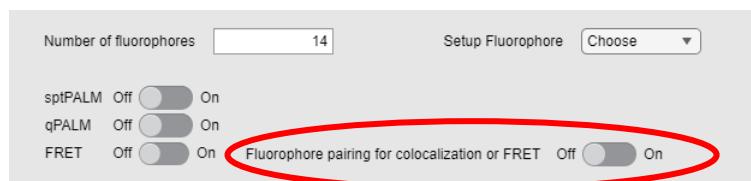
Toggling between 3D on and off will **reset the virtual samples** and some laser parameters.

Choosing the number of fluorophores (or number of colors)



Choose the number of different fluorophores in the « *Number of fluorophores* » box. The number of fluorophores is not limited, so if you want to run a 10-color experiment, you can ...

As soon as you use more than one fluorophore, you will get the option to pair fluorophores for colocalization studies.

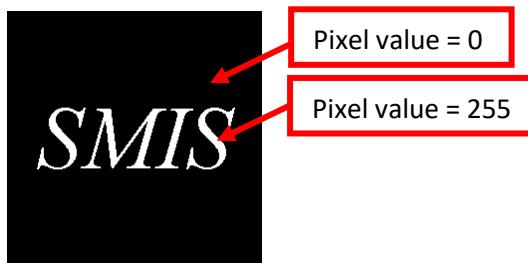


Choosing virtual samples

General characteristics of virtual samples:

Virtual SMIS samples are *segmented* images or 3D arrays. Each Pixel (2D) or Voxel (3D) will thus have a value called a *pattern ID* in SMIS. All pixels or voxels with a particular value (or pattern ID) can be considered as a *labeling target* for a fluorophore.

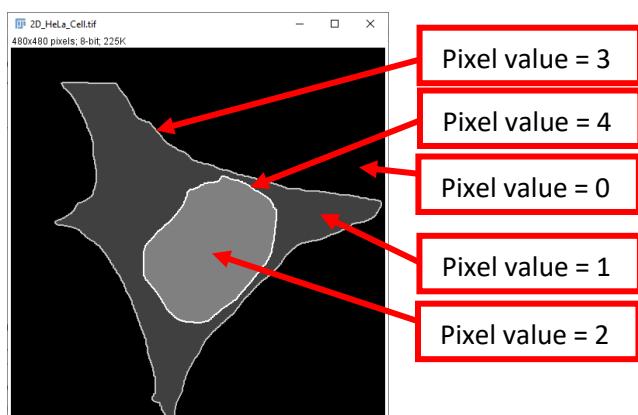
Let's take the example of the SMIS logo, a simple 2D virtual sample with only 2 IDs (0 and 255 in that particular case)



We will see later in the « *Setup Fluorophore* » how to label the SMIS logo with the fluorophores.

In this particular case if we want to label the *SMIS* pattern, we will address the fluorophores to the pattern ID = 255. Instead, if we want to label the background, we will address the fluorophores to the pattern ID = 0. Maybe, if we want to simulate unspecific labeling, we will label 95% of the fluorophores to the pattern ID = 255, and the 5% left to the pattern ID = 0.

You may have virtual samples with multiple pattern IDs, for example the one below:



In this particular case of a HeLa cell, you could address fluorophores to: the background (pattern ID = 0), the cytoplasm (pattern ID = 1), the nucleus (pattern ID = 2), the plasma membrane (pattern ID = 3) or the nuclear membrane (pattern ID = 4).



As a consequence of this, loading a nonsegmented image with multiple pixel values into SMIS will result in a big mess. So don't do that (except for qPALM samples, see below) !

For 2D simulations

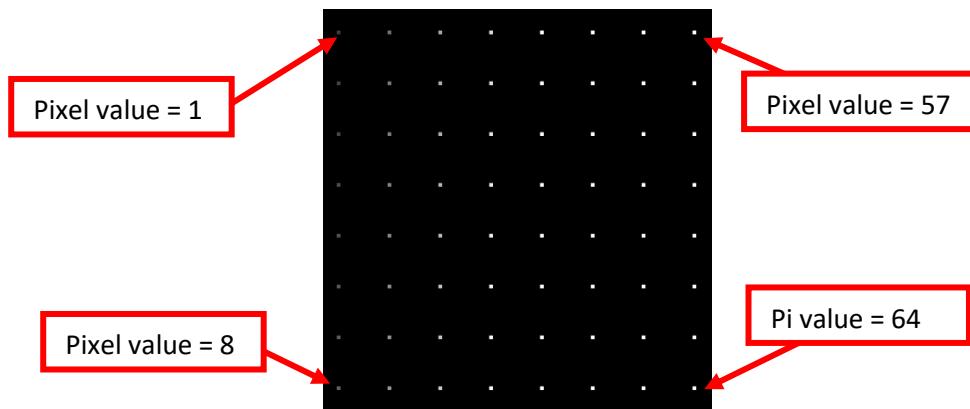
For 2D simulations, virtual samples are 2D segmented images in *.tif* format. You can define new SMIS virtual samples using for example *ImageJ*, or *Matlab*. A limited set of 2D virtual samples can also be created with the « *SMIS Create Virtual Sample tool* ».

For 3D simulations

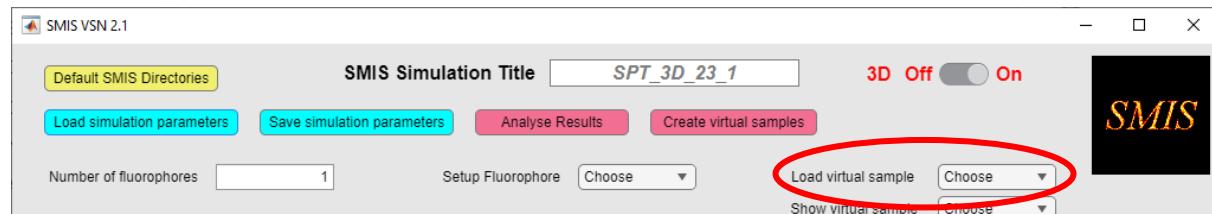
For 3D simulations, virtual samples are 3D Matlab arrays. You need *Matlab* to define new 3D virtual samples (which is presently a limitation). A limited set of 3D virtual samples can be created with the « *SMIS Create Virtual Sample tool* »

For qPALM simulations

Virtual samples for quantitative PALM simulations are typically images displaying repeated features (e.g. cluster sites) and allow placing a well-defined number of fluorophores in each of these features (i.e. defining a well-defined stoichiometry). In that case, each individual feature should be given a different pixel value, such as in the example below:

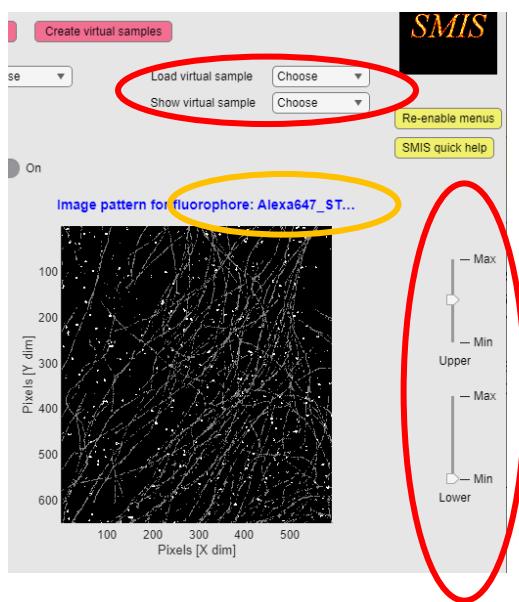
**Loading virtual samples:**

Once you have defined the number of fluorophores, virtual samples can be loaded with « *Load virtual sample* »



You can load a different virtual sample for each fluorophore in a multicolor experiment, or you can choose a single virtual sample that you will use for all the fluorophores. In case you choose different samples for different fluorophores, the virtual sample image sizes need to be the same. If a newly loaded virtual sample is a different size than previously loaded samples, SMIS will ask if you agree to reset those previous samples.

Once the samples are loaded, you can look at them with « *Show virtual sample* ».



You can adjust the contrast of the displayed virtual samples.

In principle, you can zoom in, zoom out, pan, or even rotate (3D patterns) the virtual samples, as for normal Matlab figures, by moving the mouse within the orange ellipse (above figure). However this is not very advisable, as this can be slow for 3D images, and it can mess up the SMIS interface.

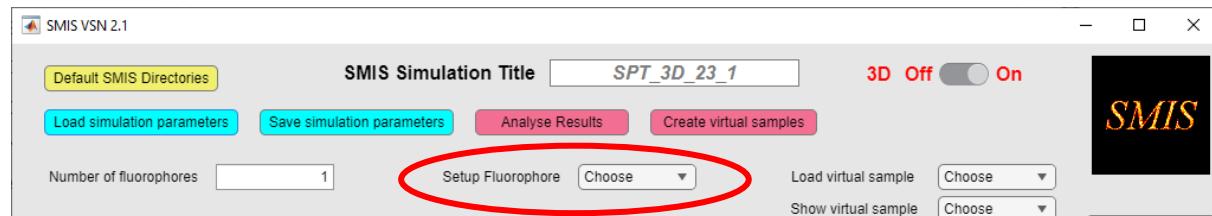
 The number of pixels in X and Y dimensions indicated on the displayed images correspond to the true size of the chosen virtual samples. The actual frame size (*Frame Size X and Y*) that will be “recorded” on the SMIS detector correspond to these dimensions divided by the « *Binning factor* », as shown on the image below.



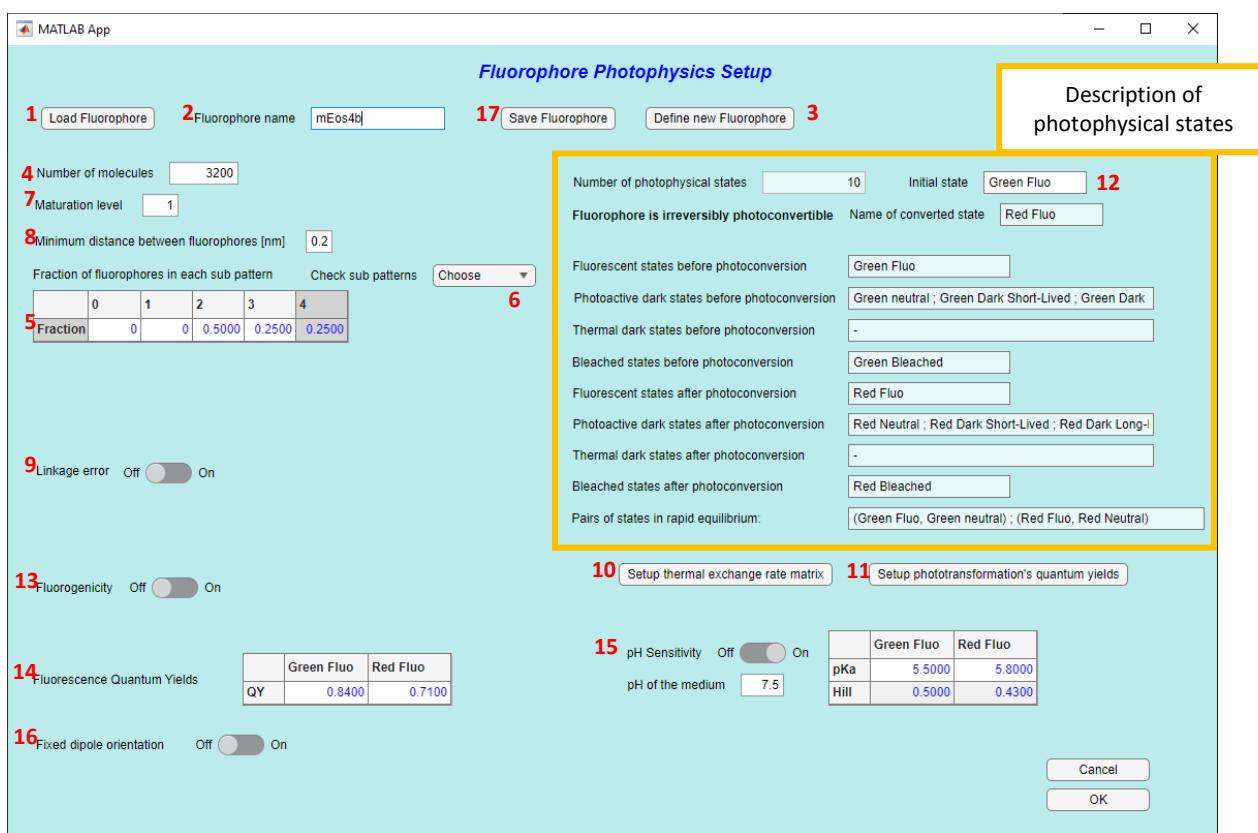
 It is thus important to realize that the “true size” of a pixel (or voxel) in the virtual sample images is equal to *Raster size / Binning factor*. Thus, changing either values will affect the dimensions of objects that will be reconstructed upon processing the image stacks. You can get a feeling of this in the « *Create virtual samples* » tool.

Setting up the fluorophores

Once the number of different fluorophores has been chosen, each fluorophore type should be set up:



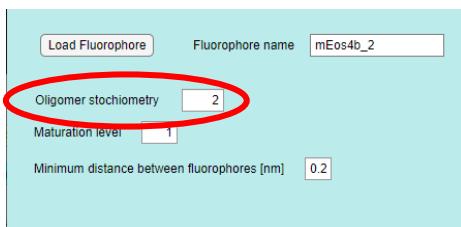
The following new window will open:



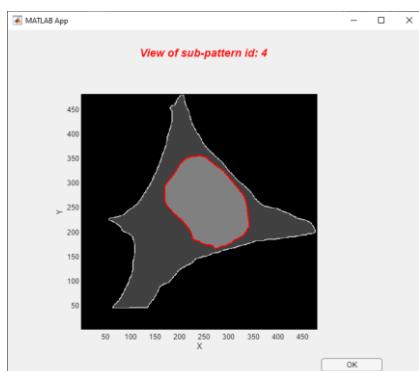
Here is what you can do, preferably in order.

1. Load a fluorophore from the SMIS fluorophore database
2. Give it a name
3. If you cannot find the fluorophore you like in the database, you can define a new fluorophore (explained below)

- Set the number of target molecules for that particular fluorophore. Note that in qPALM mode, this changes to the stoichiometry of the target (ie number of fluorophores that will be addressed to each pattern ID, see definition of virtual samples for qPALM above)



- Distribute the fluorophore through the various features (“sub-patterns”) of the virtual sample, according to their *pattern IDs* (see section *Choosing virtual samples*). In qPALM mode, this is disabled, as fluorophores will be automatically distributed.
- You can check visually which pattern IDs correspond to which feature by using « *Check sub patterns* »: for example, on the image below you can see which feature corresponds to the pattern ID = 4: this is the nuclear membrane.



- You can define a maturation level (for fluorescent proteins), which also corresponds to a labeling efficiency (for organic dyes). This corresponds to the fraction of target molecules that will be effectively labeled.
- You can also set a minimum distance between target molecules. This will ensure that target molecules will not be too close to each other.
- You can define a linkage error. This is the distance between the fluorophore and the actual target.



Be careful: a linkage error only makes sense if its value is greater than the digital resolution of the virtual sample. This is rarely the case, unless you use a highly resolved virtual sample. You can also artificially increase the binning value, or use a big linkage error, but this might not make a lot of sense. Linkage errors are not enabled in the case of FRET simulations.

As shown on the image above, if you opt for a linkage error, then you have to define the linkage length and standard deviation, and you have to choose whether this length remains constant (in length and orientation) over the whole data acquisition (typically the case for a fixed sample, but not for a live sample or in sptPALM experiments).

Also if « *Sub-pattern control* » is set to on, you can choose whether target molecules and attached fluorophores are in different sub-patterns of the virtual sample. In that case fill the "linkage target sub pattern ids" table. You might want to do this for example if the target is a membrane protein and the fluorophore should stand either inside the cytoplasm or in the external medium.



Do not use this option when the target is positioned inside a large sub pattern, because then the fluorophore would never reach the desired sub pattern. Thus, be careful if you use this option !

10. You can change interactively the rates of thermally activated transformations between photophysical states, although initial values are defined when the fluorophore is designed. For example, this is useful if you want to turn off a certain transformation.

		To				
		Fluorescent	T1	Anionic Radical	Cationic Radical	Bleached
From	Fluorescent	0	0	0	0	0
	T1	1e+05	0	200	0.2	1
	Anionic Radical	0.0025	0	0	0	0
	Cationic Radical	0.0025	0	0	0	0

11. Likewise, you can change interactively the quantum yields of light-activated transformations between photophysical states, although initial values are defined when the fluorophore is designed. Again, this is useful if you want to turn off a certain transformation, or increase it to look at the effect.

Of note, you can look at the spectra of all states in this menu.

12. Importantly, you can change the initial state of the fluorophore at the start of the simulation, by entering the *name* of the state in the text box. For example you could start a simulation with a photoconvertible fluorescent protein that is already fully convertible to the red state. If you want a fluorophore to start in a mixture of different states, you can duplicate the fluorophore and set each of the copies to a different initial starting state. Alternatively you can add an initial dummy state (using « *Define new fluorophore* ») that very rapidly equilibrate towards real photophysical states with different rates. This will allow you to populate these photophysical states differently, at the start of a simulation.
13. You can set your fluorophore to be fluorogenic. Here, fluorogenic means that the fluorophore will exhibit different quantum yield depending on the part of the virtual sample (sub-patterns) in which the fluorophore resides. This is typically useful for fluorophores that are only fluorescent when they e.g. are bound to a membrane.

Fluorogenicity	Off	<input checked="" type="checkbox"/>	On		
Level	0	1	2	3	4
	1	1	1	1	1
	1	1	1	1	1
	1	1	1	1	1

In the table, enter the fractions of the default fluorescence quantum yield corresponding to the effective quantum yields in each sub pattern of the virtual sample.

14. You can modify the fluorescence quantum yields for each fluorescent states of the fluorophore.
15. You can enable pH sensitivity of the fluorophore. This is only possible for fluorophores that are initially defined with rapidly exchanging states between protonated (nonfluorescent) and ionic (fluorescent) states. See « *Define new fluorophore* ». If pH sensitivity is enabled, then you can modify the pKa and Hill coefficients for each fluorescent state of the fluorophore, and you can also modify the pH of the medium.
16. You can control the tumbling of fluorophores. Set to On if fluorophores have a fixed dipole orientation. Set to Off if fluorophores tumble more rapidly than the acquisition rate.

If a fixed dipole orientation is chosen, it can be random for all fluorophores, or a unique

Fixed dipole orientation	Off	<input checked="" type="checkbox"/>	On				
Unique dipole orientation	Off	<input checked="" type="checkbox"/>	On	Theta [°]	0	Phi [°]	90
Allow Stochastic Reorientation	Off	<input checked="" type="checkbox"/>	On	Jump rate [s-1]	0		

dipole orientation can be chosen for all fluorophores. Fluorophores can also reorient stochastically at a defined rate.

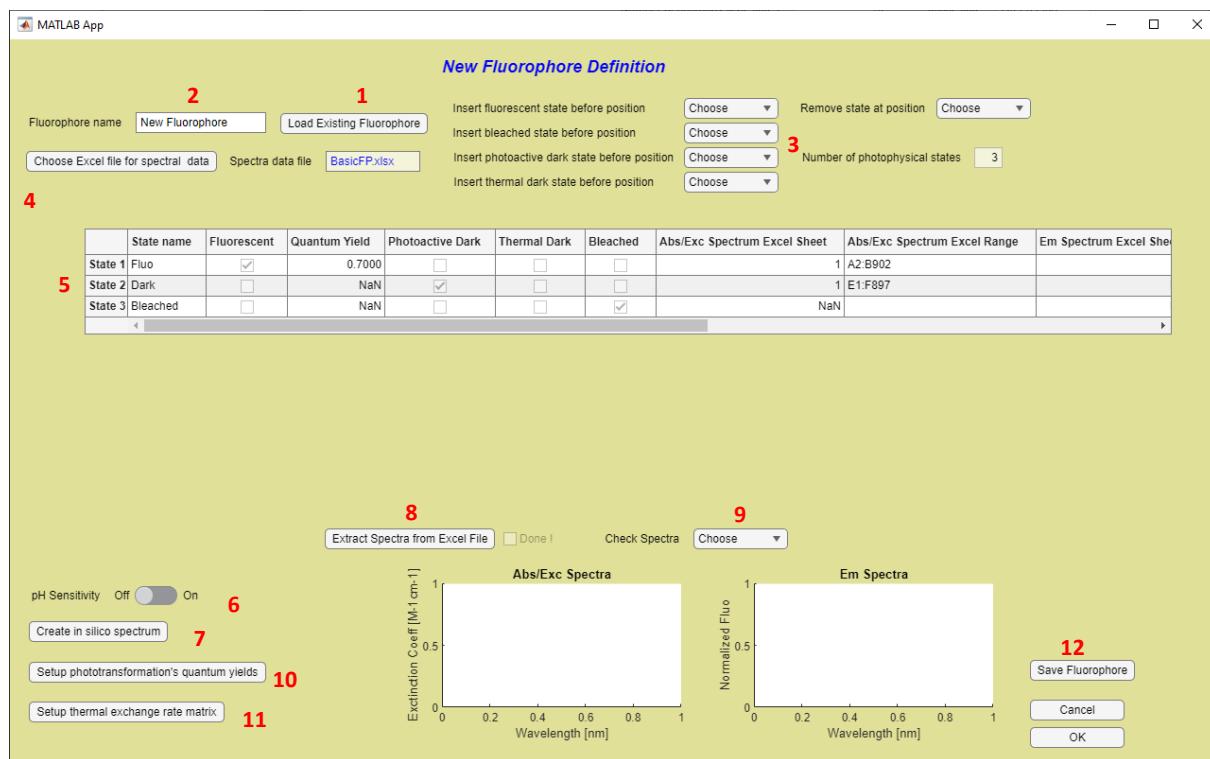
17. You can save your fluorophore at any moment. The properties that will be saved if you do so are the properties intrinsic to the fluorophore (photo-transformations, pKa's, quantum yields ...) but not the properties related to labeling (number of molecules, dipole orientation ...).
18. In multicolor experiments, you can copy all parameters from another defined fluorophore to the current fluorophore. This is in particular useful when you want to delete some fluorophores and only keep some that are at the end of the list: in that case copy the setting of the fluorophores you want to keep to those of the first fluorophores in the list.

Fluorophore Photophysics Setup

Save Fluorophore Define new Fluorophore Copy Setup From Other Fluorophore Choose ▾

Defining new fluorophores

New fluorophores can be defined in SMIS by clicking on « *Define new fluorophore* » in the *Fluorophore Photophysics Setup* window. The following window will appear:



Here is how to proceed:

1. Preferably start by loading an existing fluorophore
2. Give a name to the new fluorophore
3. Define the photophysical states by inserting fluorescent, bleached, photoactive dark states or thermal dark states, or by removing existing states. The table will be updated. Fluorescent states need to be associated to excitation and fluorescence spectra, whereas photoactive dark states need to be associated with absorption spectra. Thermal dark states and photo bleached states are not associated to any spectrum.

4. Spectra have to be defined in an Excel spreadsheet. Load that Excel spreadsheet into SMIS. Existing spectral data are stored in the SPECTRA sub-directory of the directory specified for the fluorophore's library in the “Default SMIS directories” accessed from the main SMIS window. A typical spectral data spreadsheet should look like this, but basically you can define it as you want.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	lambda	Exc	lambda	Em	lambda	Act							
2	250	0	250		0	250			0.06813559				
3	251	0.007640373	251		0	251			0.076191051				
4	252	0.015280747	252		0	252			0.0851254				
5	253	0.02292112	253		0	253			0.095025381				
6	254	0.030561494	254		0	254			0.105985224				
7	255	0.038201867	255		0	255			0.118107173				
8	256	0.045842241	256		0	256			0.131502037				
9	257	0.053482614	257		0	257			0.146289763				
10	258	0.061122987	258		0	258			0.162600038				
11	259	0.068763361	259		0	259			0.180572913				
12	260	0.076403734	260		0	260			0.200359442				
13	261	0.084044108	261		0	261			0.222122359				
14	262	0.091684481	262		0	262			0.246036755				
15	263	0.099324854	263		0	263			0.272290792				
16	264	0.106965228	264		0	264			0.301086422				
17	265	0.114605601	265		0	265			0.332640127				
18	266	0.122245975	266		0	266			0.367183669				
19	267	0.129886348	267		0	267			0.404964849				
20	268	0.137526722	268		0	268			0.446248274				
21	269	0.145167095	269		0	269			0.49131613				
22	270	0.152807468	270		0	270			0.540468947				
23	271	0.160447842	271		0	271			0.594026367				
24	272	0.168088215	272		0	272			0.653277894				

The data themselves can be extracted from popular databases such as FPbase (www.fpbase.org) for fluorescent proteins or using e.g. SpectraViewer (www.thermofisher.com/order/fluorescence-spectraviewer) for organic fluorophores. Typically, you would have to merge .csv or .txt files into a single .xlsx files suitable for SMIS. In the case of photoactive dark states, in many cases you will not find absorption spectra available in the databases. In some instances, the spectral data may be available in publications. Whenever you don't have data, you can define yourself an ad-hoc spectrum using the « *Create in silico spectrum* » tool 7. (See below)

5. Enter all relevant parameters in the table. Wherever nonrelevant, enter *NaN* (not a number). The specific location of data in the Excel spreadsheet have to be entered, ie Excel Sheet number and Excel Range, following Matlab format, as the example shown below:

	Bleached	Abs/Exc Spectrum Excel Sheet	Abs/Exc Spectrum Excel Range	Em Spectrum Excel Sheet	Em Spectrum Excel Range	Epsilon	Lambda Eps	Photoconverted	I
State 1	<input type="checkbox"/>		1 A2:B902		1 C2:D302	50000	488	<input type="checkbox"/>	
State 2	<input type="checkbox"/>		1 E1:F897		NaN	10000	405	<input type="checkbox"/>	
State 3	<input checked="" type="checkbox"/>		NaN		NaN	NaN	NaN	<input type="checkbox"/>	

For example “A2:B902” means that the corresponding data are found in columns A, rows 2 to 902 (X=wavelength) and B, same rows (Y= spectral value).

Other information to enter are:

- state name
- quantum yield of fluorescence for fluorescent states
- extinction coefficient values (Epsilon) in Mol-1cm-1
- wavelength corresponding to the entered Epsilon (Lambda Eps)
- in the case of a (non-reversibly) photoconvertible fluorophore, tick the state that is reached upon photoconversion (ie typically the red fluorescent state for green to red photoconvertible fluorescent proteins)
- if pH sensitivity is set (6.), tick the states that are in rapid equilibria, typically for fluorescent proteins anionic chromophore states (fluorescent) in equilibrium with protonated states (dark photoactive).

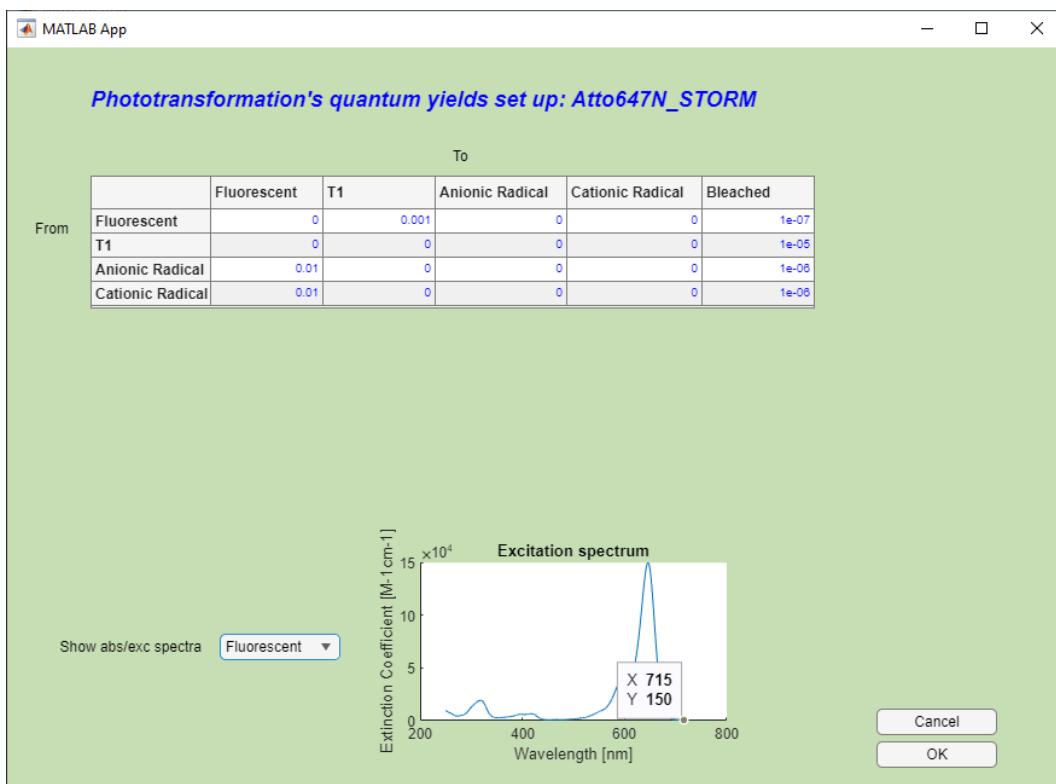
 A limitation in the current version of SMIS is that states in rapid equilibria are limited to fluorescent states in equilibrium with nonfluorescent states, and they should be defined for all fluorescent states (ie green and red for a photoconvertible fluorescent protein).

- If pH sensitivity is set (6.), also enter pKa and Hill coefficients for all the states in rapid exchange.

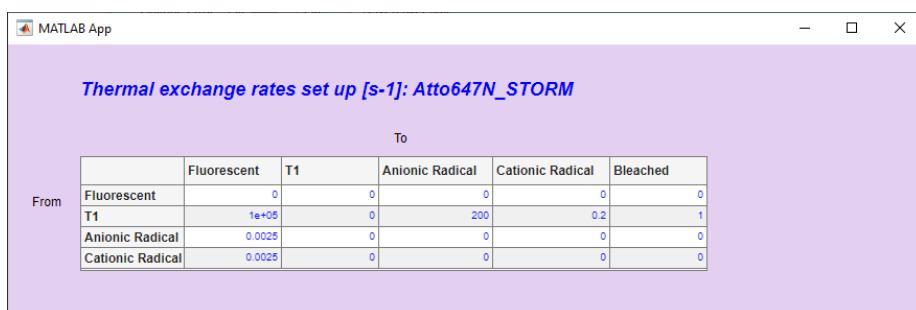
The best is to look at examples available in the SMIS fluorophore database.

6. Use this to set pH sensitivity on or off
7. Use this to create your own spectrum, see below.
8. Once this is done, extract the spectra from the Excel spreadsheet.
9. And check them

10. Then setup phototransformation quantum yields according to the knowledge available for the new fluorophore. If you don't know anything about those, you can use the values defined for the closest fluorophore in the SMIS database. You may want to tune the values qualitatively, and see what the effects are on simulated data. You may want to compare your output simulated data with experimental data, for example photobleaching curves.



11. Do similarly and in 8. For the thermal transformation rates between photophysical states

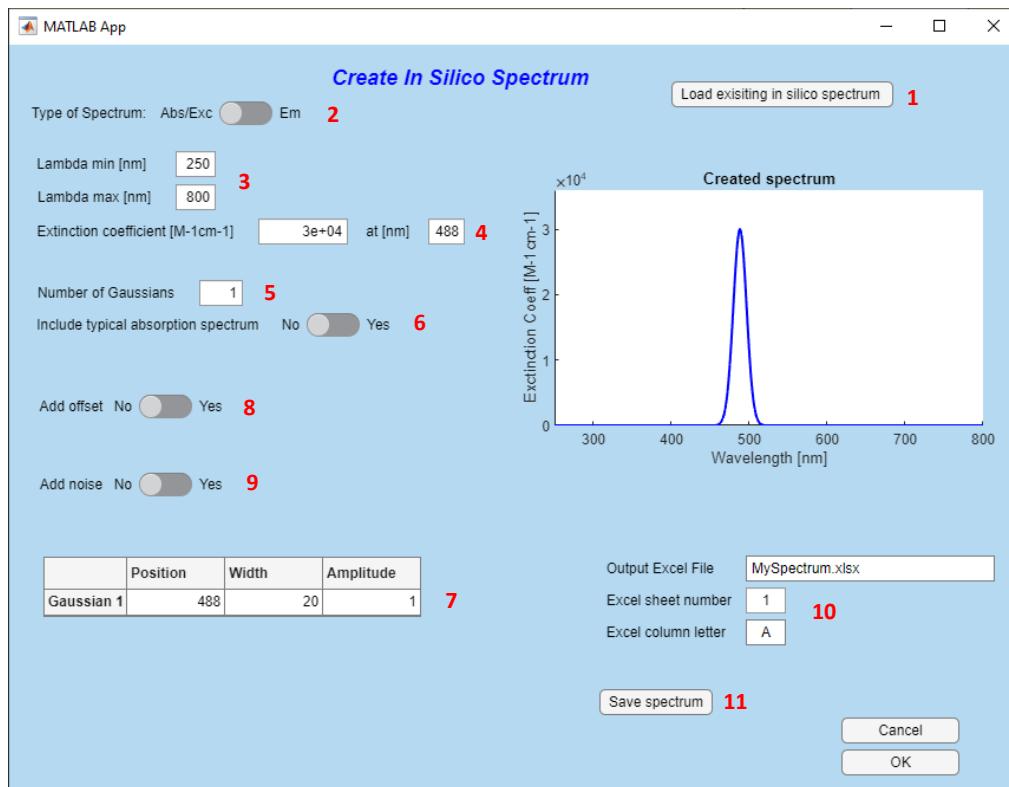


12. Once everything is completed, save the fluorophore in the SMIS database.

Creating spectra

You can create spectral data in SMIS, in case those are not available in forms of hard-core data. Maybe you have available spectra in forms of figures in publications, or maybe you have some general knowledge such as for example the fact that protonated chromophores in green fluorescent proteins generally absorb at around 400 nm with a Gaussian-like looking band.

To create such a spectrum go in « *Create in silico spectrum* » tool from the « *Define new fluorophore* » window.



Using this tool, you can create absorption/excitation or emission spectra.

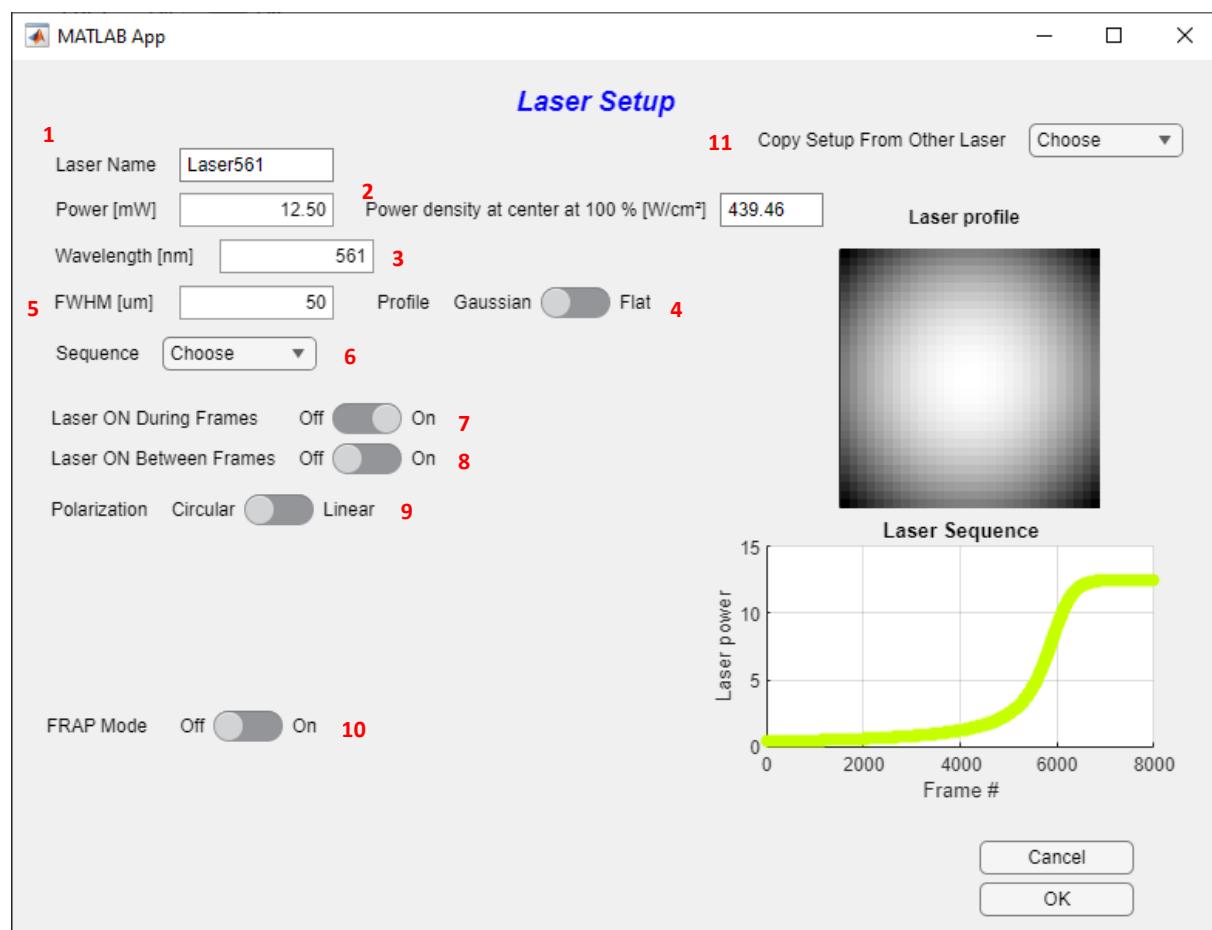
1. Optionally load an existing spectrum, if there is one which you know looks close to the one you want to define.
2. Decide whether you will define an absorption/excitation or an emission spectrum.
3. Define λ_{\min} and λ_{\max}
4. Set the extinction coefficient at a defined wavelength
5. Decide how many Gaussian's you need to describe your spectrum
6. In addition or alternatively, decide if you want your spectrum to include a “typical” absorption/excitation or emission shape.
7. Tune the position, width and amplitude of all the components
8. Eventually add an offset, which allows you to force the extinction coefficient to have a defined value at a certain wavelength typically far from the peak
9. Possibly add noise, but this is not very useful.
10. Define the output Excel file name, spreadsheet number and column letter.
11. Finally save the spectrum

Setting up lasers

Once fluorophores and virtual samples have been defined, lasers should be set up. You can define as many lasers as you want.



To set up a laser, use « *Setup laser* » in the main SMIS window. You'll get the following window:



12. Give a laser name (usually containing the chosen wavelength)
13. Choose the laser power, or the laser power density at beam center. These 2 values are related, depending on the beam shape as defined in 4. and 5.
- !** A virtual sample has to be loaded to get the power density. (This is because the virtual samples define the image size).
14. Choose the wavelengths. The color of the laser sequence plot will match the chosen wavelength.
15. Choose the type of profile: Flat or Gaussian beam.
16. Choose the width of the beam
17. Choose the laser sequence along that acquisition (see below for examples). The currently used laser sequence is shown on the plot on the lower right of the window.



For some specific laser sequences (e.g. Fermi profile) which are dependent on number of frames in the acquisition stack, the sequence might be automatically reset if you happen to change the number of frames after you define a sequence.

18. Decide whether the laser is *on* during frame time
19. Decide whether the laser is *on* in between frames. You can only do that if the time between frames defined in the main SMIS window is > 0
20. Set laser polarization. The laser can have circular or linear polarization. If linear polarization is chosen, define the angle of the laser polarization relative to the X axis of the focal plane.

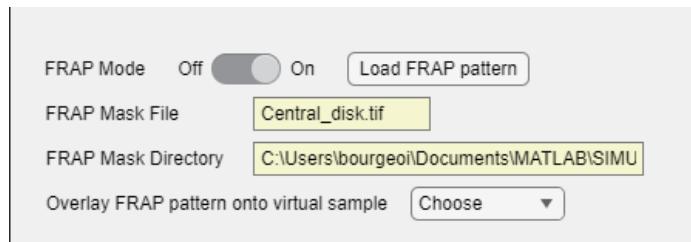


Laser polarization is only relevant when the option of anisotropic fluorophore dipole orientation is chosen for at least one fluorophore.

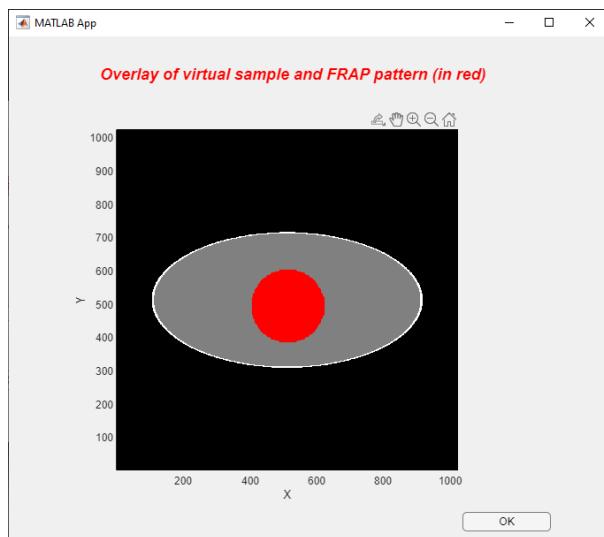
21. Lasers can be used in a “FRAP mode”. This means that the laser will only be applied in a restricted field of view defined by the FRAP pattern to be loaded. This is useful to simulate FRAP or pulse-chase experiments.



The FRAP pattern image must be an image of the same dimension as the camera image size defined in the main SMIS window.

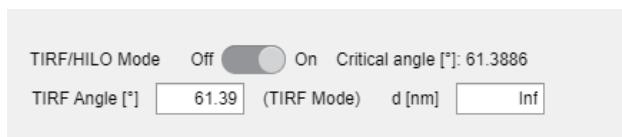


To make sure, the FRAP pattern is correctly positioned on your virtual sample, use « *Overlay FRAP pattern onto virtual sample* »



22. You can copy all parameters from another defined laser to the current laser. This is in particular useful when you want to delete some lasers and only keep some that are at the end of the list: in that case copy the setting of the lasers you want to keep to those of the first lasers in the list.

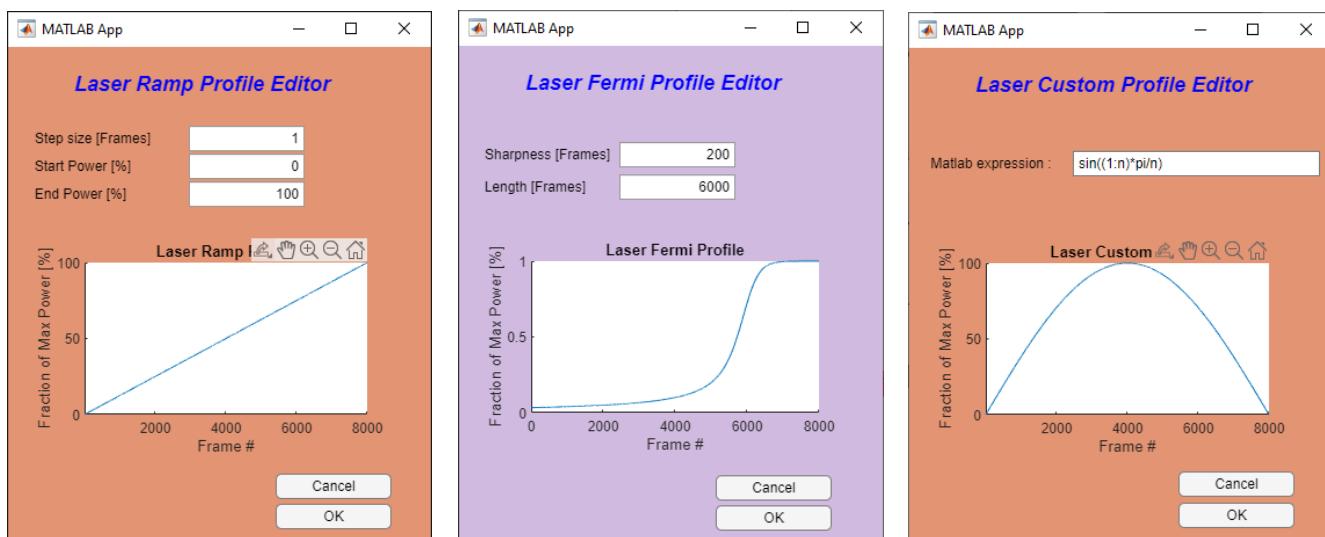
23. In 3D mode, you have the option to use lasers in HILO or TIRF mode



Depending on the angle that you enter (lower or higher than the critical angle corresponding to the objective specifications (see « *Objective and PSF* » from the main SMIS menu), shown on the window), the TIRF or HILO mode will be selected. The characteristic depth d probed by the laser beam is then shown. Alternatively, you can change this characteristic depth, and the angle will be set accordingly.

Lasers sequences

Various types of laser sequences (evolution of laser power along data acquisition) can be entered. Three are shown below:



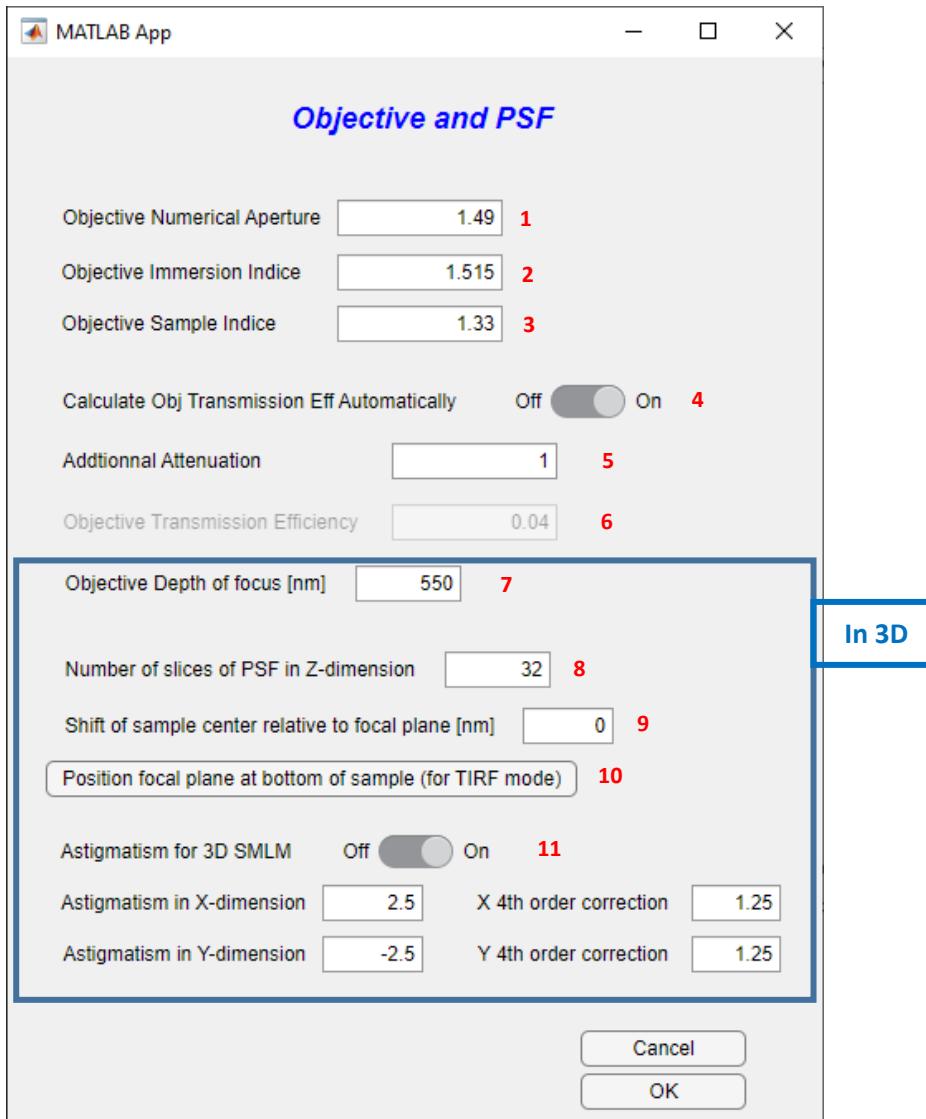
For the custom laser profile editor, a mathematical expression in Matlab format has to be entered. In this expression n is the number of frames.

For example, to get a single pulse at frame number 4 (e.g. for a FRAP experiment), use:

$$(1:n)>3 \& (1:n)<=4.$$

Setting up Objective and PSF

Upon selecting « *Objective and PSF* » in the main SMIS window, the following window will appear:



1. You can enter the objective numerical aperture
2. The immersion medium refraction index
3. The sample medium refraction index
4. You can either set the objective transmission efficiency manually, or automatically calculate it based on the formula of Fourkas, 2001, Opt.Letters which is recommended.
5. In the latter case, you can add an additional attenuation that takes into account the additional losses due to filters, etc. The total microscope transmission efficiency will then be the automatically calculated number multiplied by the additional attenuation.
6. In the former case, enter the total transmission efficiency of the microscope.

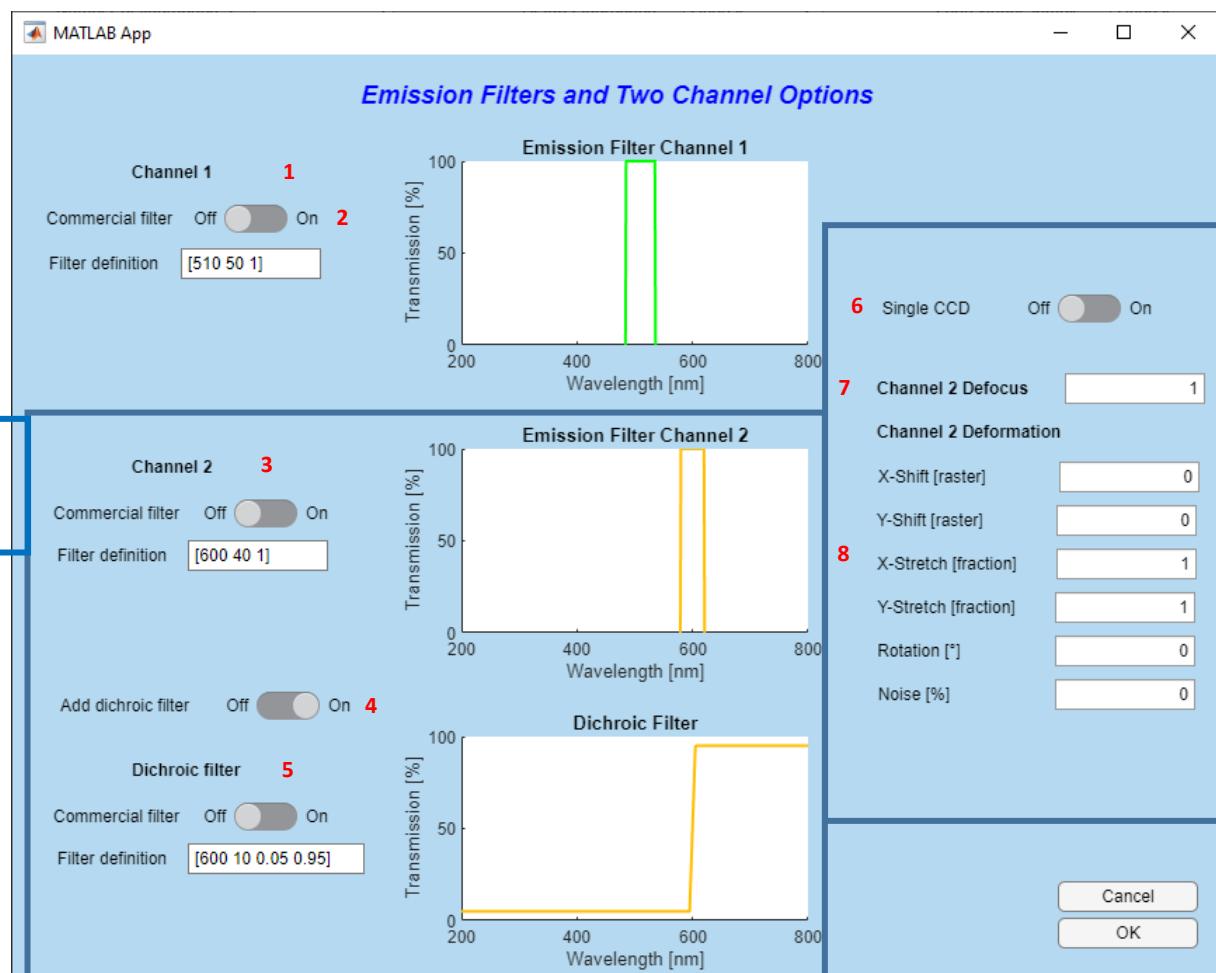
The rest of the menu only appears in 3D mode

7. You can set the objective depth of focus

8. This number basically will define the resolution of the 3D PSF that will be calculated. The higher the number, the more accurate but the more computing time.
9. This defines the positioning of the virtual sample relative to the focal plane of the objective. If set to zero, the center of the virtual sample in the Z dimension is put at the focal plane.
10. Yes this to set the focal plane at the bottom of the sample so that its bottom surface is in focus, which is what is wanted for TIRF imaging notably.
11. Use this switch to perform 3D imaging via astigmatism. In that case, you have to define the extent of astigmatism as well as the fourth order corrections in the X and Y dimensions

Setting up Emission Filters and Acquisition Channels

Upon selecting « *Emission Filters and Acquisition Channels Options* » in the main SMIS window, the following window will appear:



1. An emission filter for channel 1 can be defined here. If a commercial filter definition is not used (2.), then use Matlab format to define emission filter. You can define for example a single band filter: e.g. [500 540 0.8] where 500 nm is the low-limit wavelength, 540 nm is the high-limit wavelength and 0.8 is the filter transmission. You can also use the format: e.g. [520 40 0.8] where 520 nm is the central wavelength, 40 nm is the bandpass and 0.8 is the filter transmission. To define a long pass filter use e.g. [700 inf 0.6]. You can also define multiband

filters using e.g., for a dual band filter [[500 540 0.8]; [700 40 0.6]].

Importantly, you can check how your defined filter fits to the employed fluorophores by checking « *Show fluorophore spectra* » in the main SMIS window

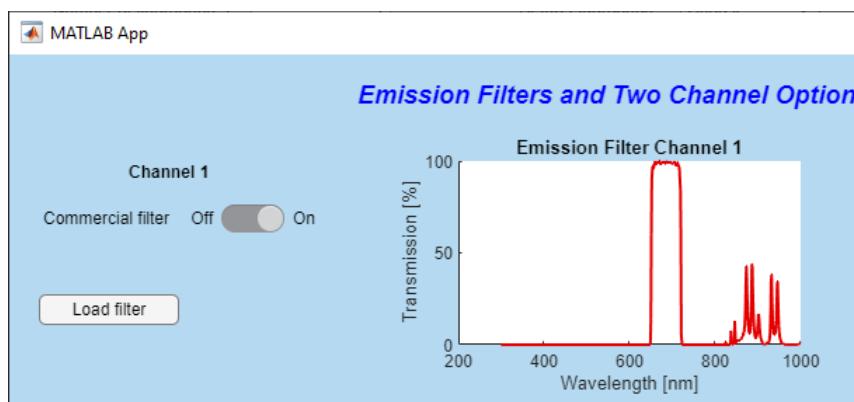
2. Alternatively you can also import a commercial filter, using text file format downloaded from e.g. Chroma <https://www.chroma.com/> or Semrock <https://www.semrock.com/>. The text file can have a header and should look like for example this:

```

SEMROCK_DICHROIC_FF545_650-Di01_Spectrum.txt - Bloc-notes
Fichier Edition Format Affichage Aide
Typical Measured Spectrum of Semrock FF545/650-Di01 Optical Filter
For wavelengths between 320 and 1120 nm, values near or below 3e-7 (Optical Density 6.5) are measurement noise limited.
For wavelengths < 320 nm and > 1120 nm, values near or below 3e-6 (Optical Density 5.5) are measurement noise limited.
Data format: Wavelength (nm) <tab> Transmission (0 to 1)
300.0 2.3493e-05
300.2 2.3603e-05
300.4 2.3714e-05
300.6 2.3825e-05
300.8 2.3937e-05
301.0 2.4049e-05
301.2 2.4161e-05
301.4 2.4275e-05
301.6 2.4388e-05
301.8 2.4503e-05
302.0 2.4617e-05
302.2 2.4733e-05
302.4 2.4849e-05
302.6 2.4965e-05
302.8 2.5082e-05

```

When importing a commercial filter, you will get something like this:



When you click on *Load filter*, by default you are directed to the (relatively small) database of filters already available in SMIS.

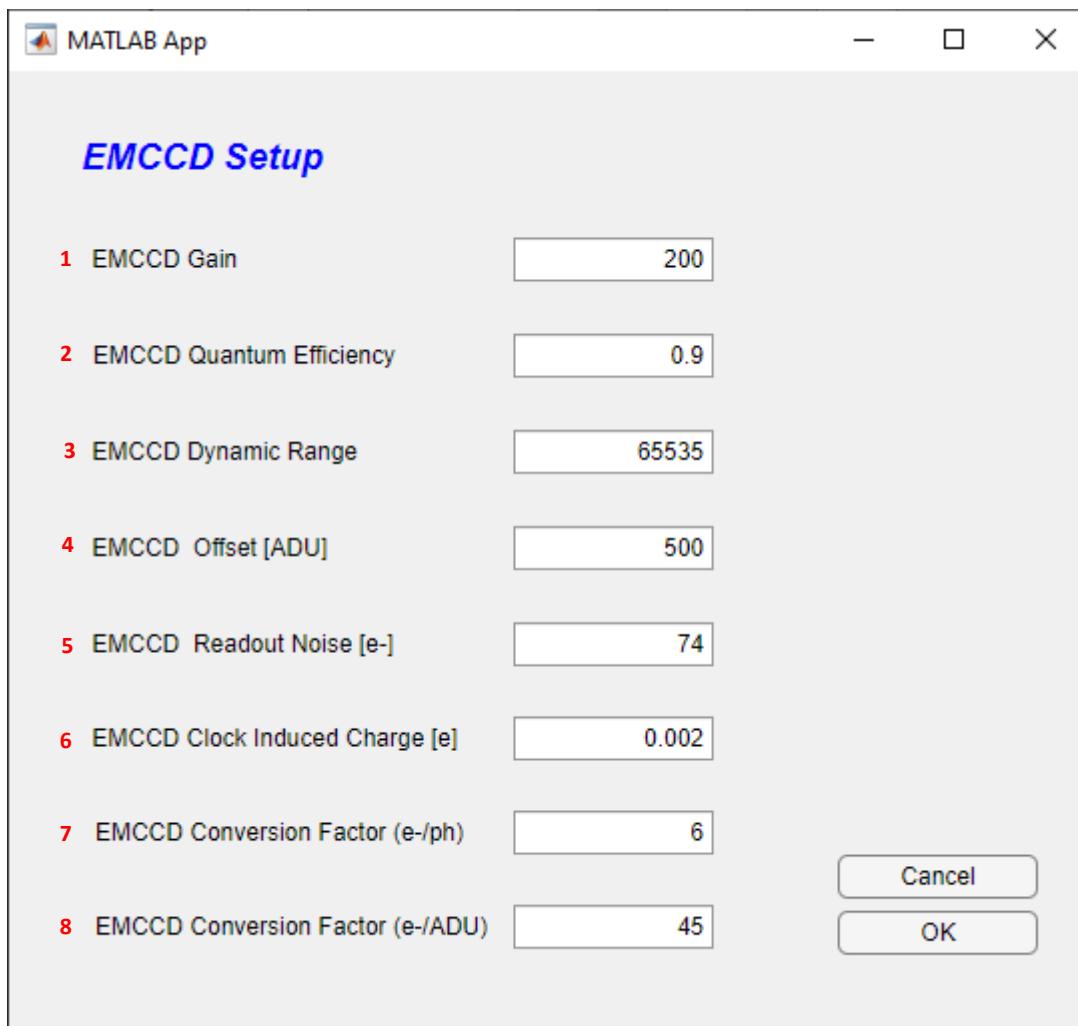
The rest of the window only appears in dual channel mode. You can set this mode in the main SMIS window using « *Number of acquisition channels* ».

3. Similarly as for Channel 1, enter filter definition for Channel 2
4. You can optionally also enter a dichroic filter.
5. If a dichroic filter is chosen, similarly as for the emission filters, this can be based on a Matlab formula or on importing a commercial filter. When using a Matlab formula, use e.g. [600 10 0.1 0.9] where 600 nm is the cutoff wavelength, 10 nm is the transition bandpass, 0.1 is the low-pass transmission and 0.9 is the long-pass transmission.

6. For multichannel experiments, you can simulate either a 2-camera mode (« *Single CCD* » set to *off*) or a single-camera mode (« *Single CCD* » set to *on*). In the former case two *.tif* stacks will be produced, one for each camera, and in the latter case a single *.tif* stack will be produced where the images of the two channels are adjacent (equivalent of “split view”).
 7. The two channels may not be both perfectly in focus. Set here the defocus of the second channel relative to the first. The PSF width will be multiplied by the entered value in the second channel.
-  This is however only valid currently for 2D simulations.
8. To simulate effects such as chromatic aberration, the second channel might be deformed relative to the first, through X and Y shift, X and Y stretch, and image rotation. A certain amount of noise in these deformations can be introduced.

Setting up the EMCCD camera

Upon selecting « *EMCCD camera* » in the main SMIS window, the following window will appear:



1. Define here the EMCCD gain : the signal (and clock induced charge noise) will be amplified by this factor before detector readout.
2. Define here the EMCCD quantum efficiency. If for example set to 0.9, only 90% of the photons will be detected.

3. Define here the EMCCD dynamic range (typically 16 bits)
4. Define here the EMCCD offset (number of ADU's recorded per pixel in the absence of any signal)
5. Define here the EMCCD readout noise in electrons : this is the electronic noise that is induced by the reading of the detector (independent of EMCCD gain).
6. Define here the EMCCD clock induced charge in electrons. This is an electronic noise in CCD detectors that adds up to the input signal and that will be amplified together with it depending on the set EMCCD gain
7. Define here the EMCCD conversion from photons to electrons (ie : number of electrons produced by one photon)
8. Define here the EMCCD conversion factor from electrons to ADU (ie : number of electrons to produce one ADU).

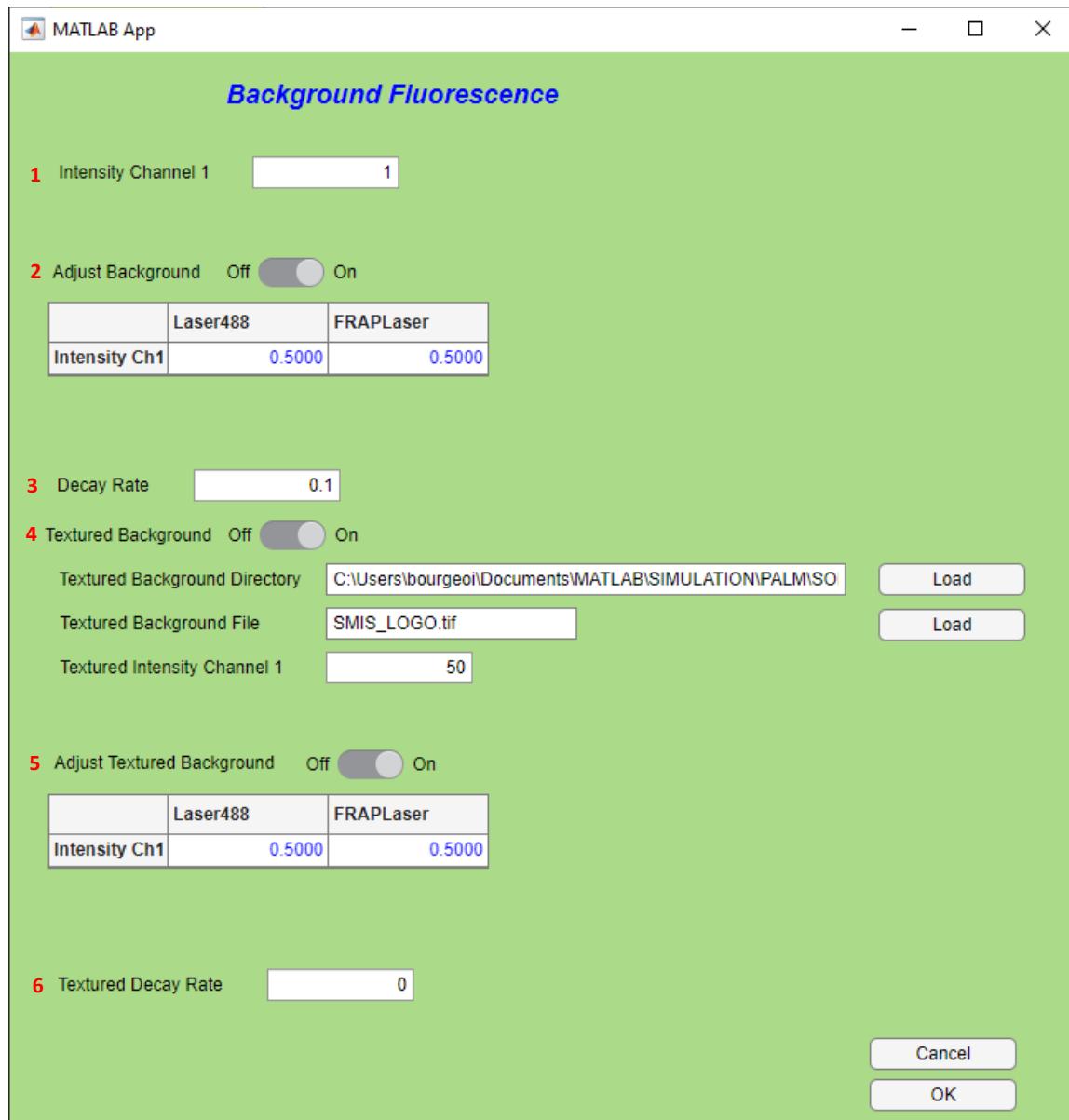
If you use Thunderstorm to process the data, you will need to set up the camera parameters according to the parameters above.



Note that in thunderstorm the parameter named "*photoelectrons per A/D counts*" corresponds in SMIS to the ratio "*EMCCD conversion factor (e-/ADU)*" divided by "*EMCCD conversion factor (e-/ph)*", giving in fact an EMCCD conversion factor in (ph/ADU) which is abusively named *photoelectrons per A/D counts* in Thunderstorm.

Setting up the fluorescence background

Upon selecting « *Fluorescence background* » in the main SMIS window, the following window will appear:

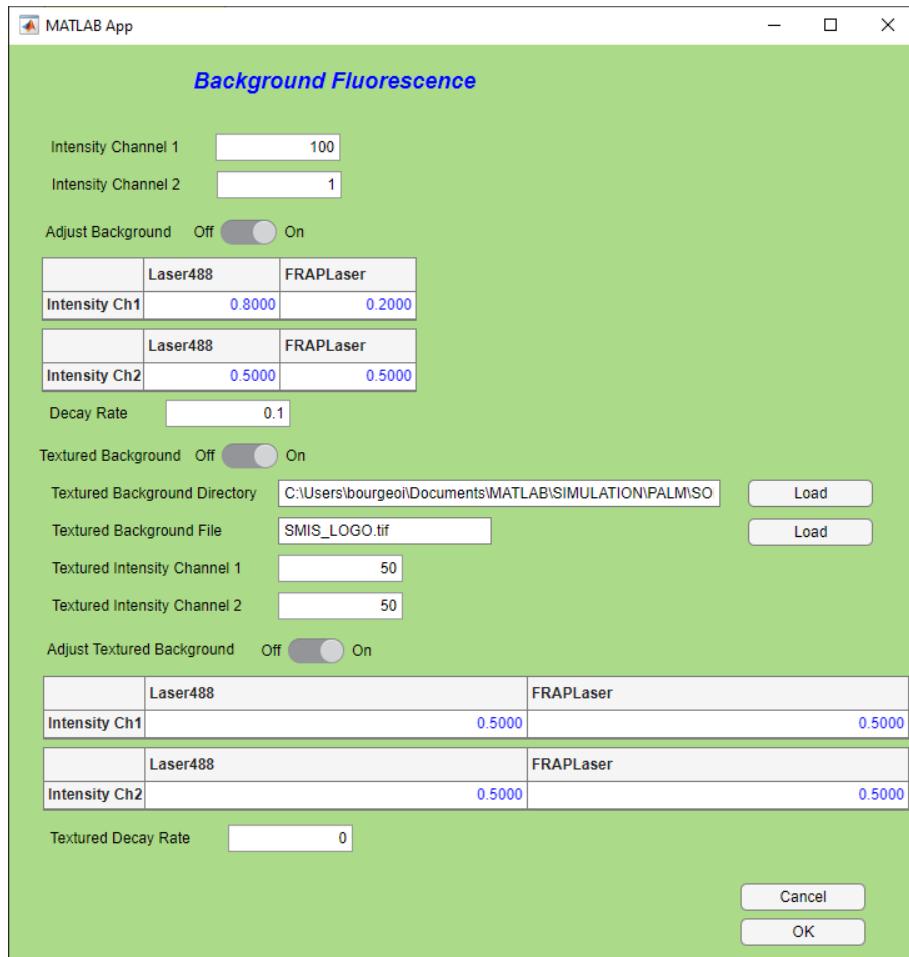


1. Define the background Intensity in Channel 1. The unit is [photons/100x100nm²/100W/cm²/s], so that the background is given per unit area and is sensitive to laser power and time. Thus, if you change laser power or e.g. frame time, the background will be changed accordingly.
2. Moreover, you can adjust the background according to local laser power density, ie e.g. for a Gaussian laser beam, you will have more background at the center of the images. Furthermore, you can adjust how background depend on the different lasers that you are using, putting more weight for example on green lasers than red lasers.
3. The background may bleach over time, which is usually the case. The decay rate is specified in [uJ/100x100nm²/s], so it also depends on employed laser powers.
4. In addition to the homogeneous background, you can define a background pattern in SMIS, according to a specific image, which is usually the same as the virtual sample in 2D or a z-projection of the virtual sample in 3D. If you select this option, enter the directory file name and intensity of the textured background in the same unit as above:

[photons/100x100nm²/100W/cm²/s]. Note however that if the image used for textured background has several pattern IDs (ie pixel values > 0), the textured background will be applied to all pattern IDs in the same way.

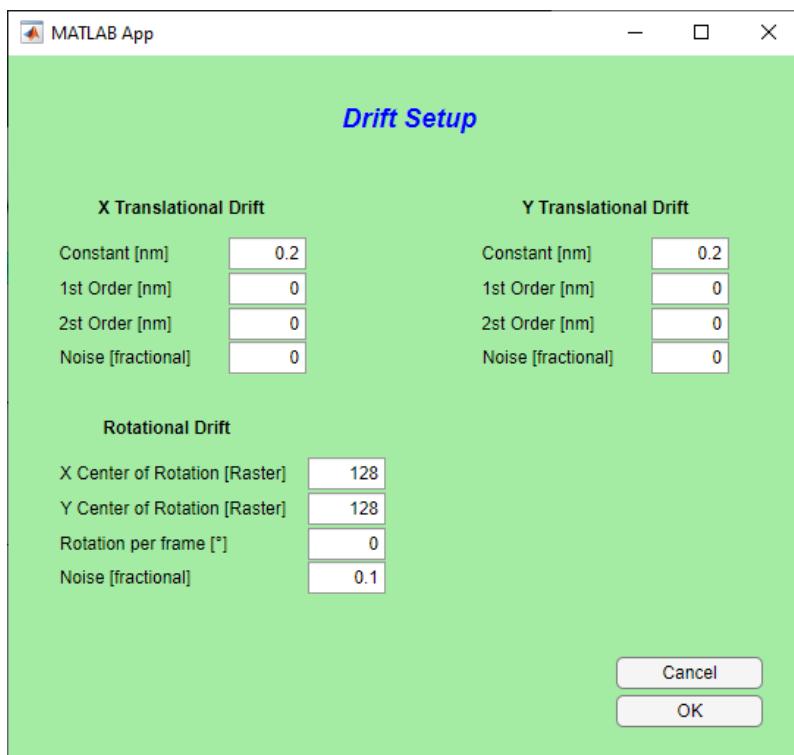
- 5. Similarly as in 2., you can adjust the textured background according to local laser intensity
- 6. Similarly as in 3., you can set a decay rate for the textured background

All these features are duplicated in the case of a 2-channel experiment:



Setting up sample drift

Upon setting « *Sample drift* » to *On* in the main SMIS window, you can enable sample drift during data acquisition. Then, by pressing on « *Edit drift* », the following window will appear:



You can set translational drift in the X and Y dimensions, and in the Z dimension for 3D simulations. For each dimension, you can define a constant, a 1st-order and a 2nd-order drift per frame. Some noise can be added, as a fraction of pixel size. The formula to calculate the drift d is then:

$$d \text{ [nm]} = \text{Constant} + \frac{fn}{N} \times \text{1st order} + \left(\frac{fn}{N} \right)^2 \times \text{2nd order} + \text{Noise/pixelsize}$$

With fn the frame number and N the number of frames in the simulation.

Rotational drift can also be added in the X and Y plane: you can define a center of rotation, a rotation per frame and some noise.

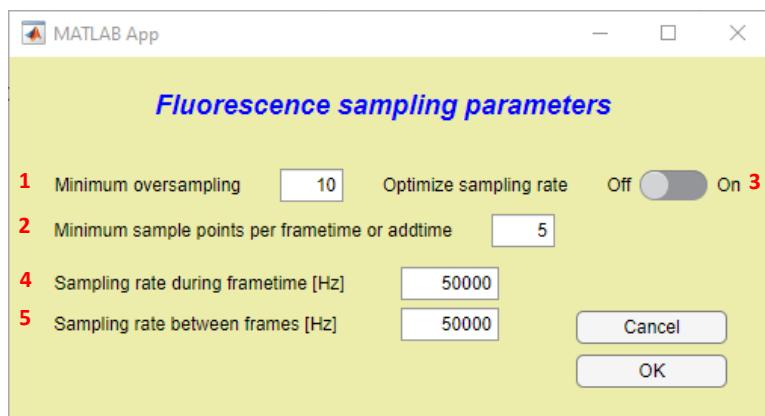


It is not advisable to set any drift in sptPALM experiments, as the drift will make molecules move but not the virtual samples, so that confinement and exchange between diffusion states might be corrupted.

Tip: You can use drift artificially, for example to scan the sample in Z when you want to calibrate the PSF for 3D astigmatism.

Fluorescence sampling rate

Upon selecting « Fluorescence sampling rate» in the main SMIS window, the following window will appear.



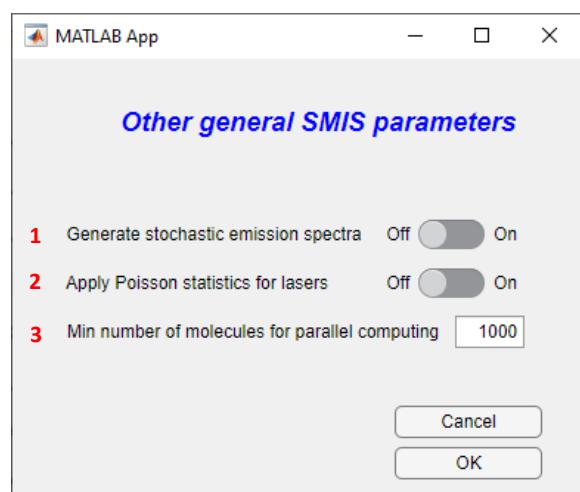
1. Minimum oversampling of fluorescence photophysical states relative to the maximum transformation rate experienced by the fluorophores. By default, a value of 10 is set, but this can be decreased down to 2 or 1 in case of very long simulation times, and depending on the goal of the simulation.
2. On the other hand, even if photophysical states change really, you want to set the minimum number of sampling points per frametime or in between frames.
3. The effective sampling rate can be optimized to match the specified minimum oversampling and minimum sample points per frametime or addtime. **This is in general advisable.**
4. In case optimization of sampling rates is not chosen, the sampling rate during frametime can be entered manually. Note that the input sampling rate might be raised automatically by SMIS if imposed by the calculated photophysical transformation rates. However SMIS will not lower the entered rate.
5. Same for the sampling rate during addtime.



Proper setting of the fluorescence sampling rate can drastically affect simulation time. A too high sampling rate may not be useful and will slowdown computing, whereas a low sampling rate may affect accuracy.

Other general parameters

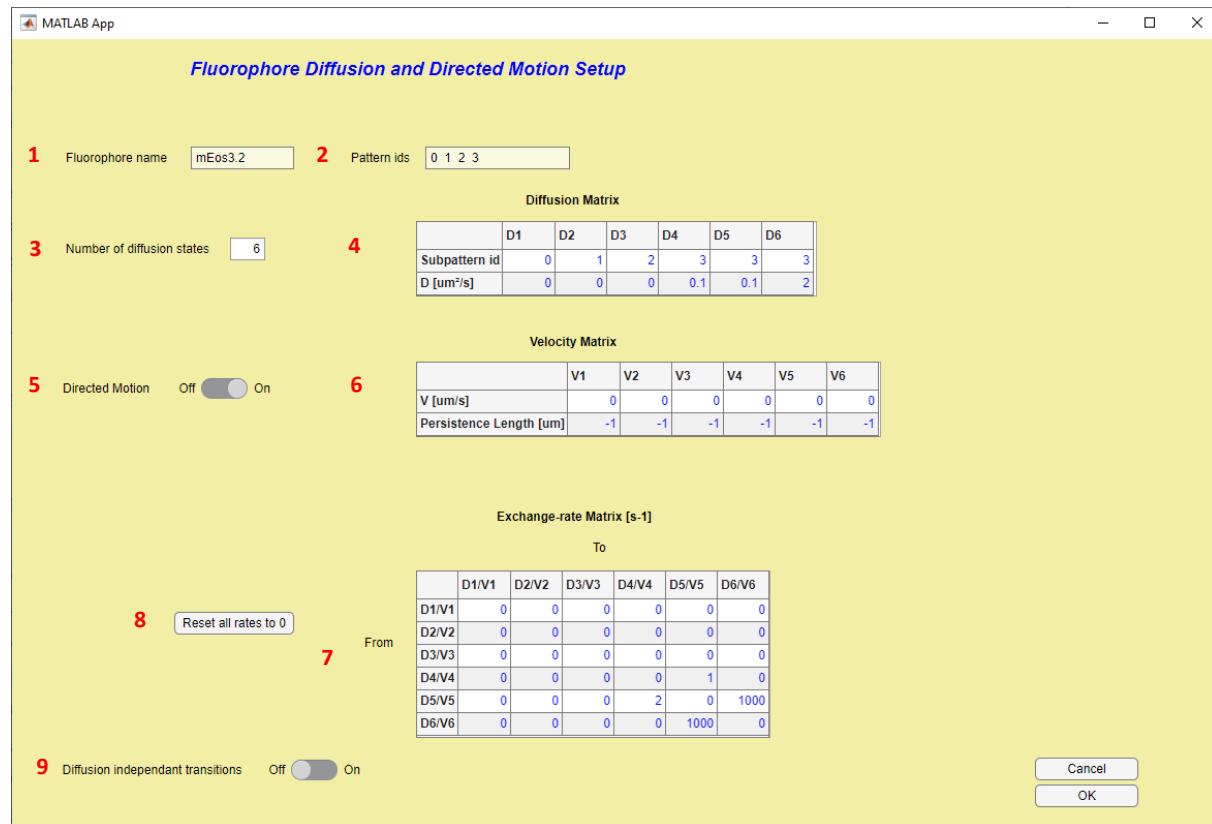
Upon selecting « *Other general parameters* » in the main SMIS window, the following window will appear:



1. If this option is turned on, stochastic emission spectra will be calculated for each non-bleached molecule at every frame. For a high number of emitted photons, these spectra will resemble the ensemble emission spectrum of the fluorophore input in the fluorophore definition. This option generates a lot of data and should not be used for simulations involving many molecules and/or many frames.
2. If this option is turned on, the number of photons absorbed by the single molecules upon laser illumination will be Poisson-distributed instead of calculated in a deterministic manner. As usually this number is quite high, this is really useful, but it might be considered in case of very low illumination regimes.
3. This is the minimum number of molecules required to allow turning on parallel computing: below this number, even if parallel computing is set on, it will be turned off because it will be very inefficient. During a simulation starting with a high number of molecules, when the number of unbleached molecules will become below the entered minimum number, parallel computing will be automatically switched off.

Single Particle Tracking

Upon setting « *sptPALM* » to *on* in the main SMIS window, you will be able to set up diffusion and directed motion parameters for each fluorophore. Select the fluorophore you want to address and the following window will appear:



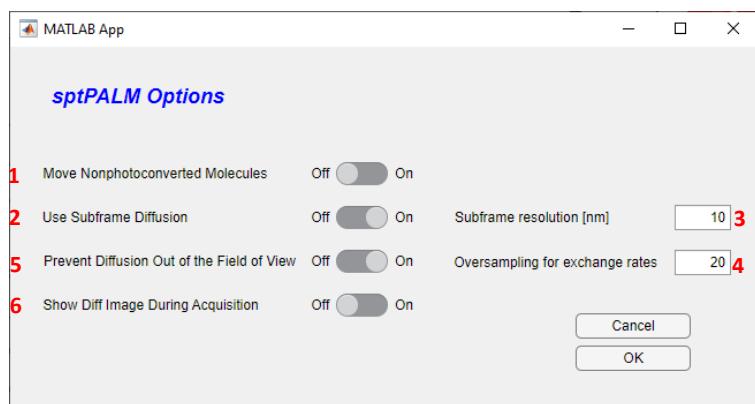
6. The name of the fluorophore is indicated here
7. The different *pattern IDs* of the virtual sample associated to that fluorophore are also indicated (see chapter on Virtual samples)
8. Choose the number of diffusion states that the fluorophore can have. Note that a minimum of 1 diffusion state should be defined for each pattern ID in the virtual sample.
9. Fill up the corresponding Table. For each diffusion state, give the value of the diffusion coefficient and the pattern ID where the single molecules may take this diffusion state. You can set multiple diffusion states for a single pattern ID, ie single molecules may change their diffusion states within a single compartment of the virtual sample.
10. In addition to diffusion, you may add directed motion, according to the same principle.
11. Fill up the corresponding Table. For each diffusion state, give the value of the velocity and corresponding *persistence length*: the persistence length indicates the typical distance over which the pattern ID does not change its orientation, that is the distance over which molecules should travel in a rather straight manner. Use a value of -1 for automated determination of the persistence length. A long straight filament should have a long persistence length whereas the bulk of the cytoplasm of a cell for example should rather have a short persistence length.
12. Can change their diffusion state according to defined kinetics. If this is the case, indicate the exchange rates in this table. This can be from one subpattern to another, or within a single subpattern.

13. You may reset all rates to 0 if needed.
14. You may want to select this option to force a possible transition to a new pattern ID even if the molecule did not move to that pattern. This mimics molecular unbinding in e.g. PAINT experiments, where the unbinding rate is *not* diffusion dependent. Thus, this option should only be used for patterns IDs that are very thin, otherwise it does not make sense, as the molecules will jump a long way. If you select this option, fill up the table to decide to which diffusion states the option should be applied:

Diffusion independant transitions	Off	<input type="checkbox"/>	On					
		Force transition	D1	D2	D3	D4	D5	D6
			<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

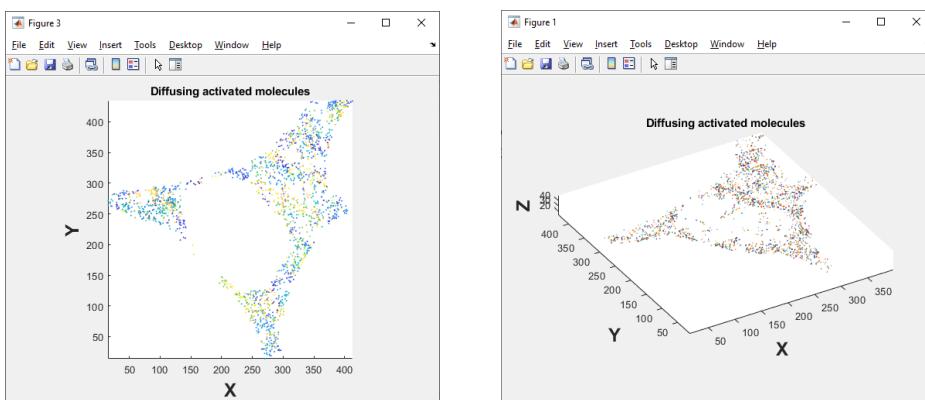
sptPALM options

Upon selecting « *sptPALM options* » in the main SMIS window, the following window will appear:



1. Set this switch to *On* if the fluorophores that have not yet been photoconverted are to be moved according to the diffusion states. As these molecules are typically not visible, this represents potentially non-useful extra calculations. Of course this only makes sense if you are using a photoconvertible fluorophore !
2. Use this option to perform subframe diffusion calculations. This is more realistic and should be the preferred option but represents a significant increase in computing time. As a consequence of this, the PSF of the moving molecules may appear smeared according to the motion that occurs during frametime
3. If *Use subframe diffusion* is chosen, this value sets the resolution (basically the distance a molecule must travel for a new calculation to be performed) for the subframe diffusion calculation. The smaller the value the better, at the cost of increased computing time.
4. If *Use subframe diffusion* is chosen, this value sets the oversampling factor for exchange rates between different diffusion states. It is possible to use a value << 1 (ie you do not care when a diffusion change occurs within the frame time) to speed up calculations e.g. for PAINT simulations.
5. Use this option to prevent moving molecules to move out of the field of view. This can be useful for example if molecules are diffusing in the background and you don't want to "lose them" to keep a fixed effective concentration, e.g. in PAINT.

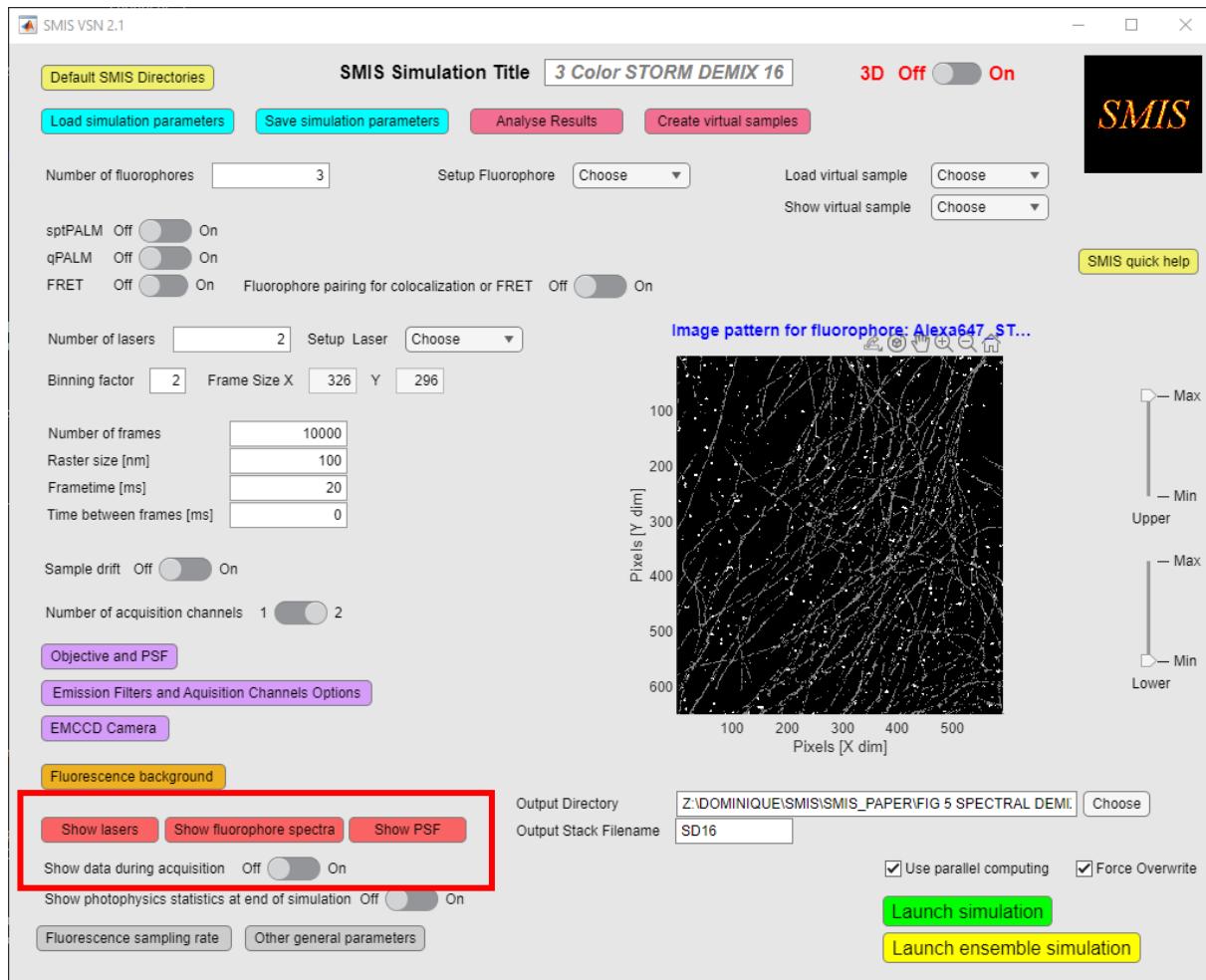
6. Use this option to visualize moving molecules in an extra Matlab figure during data acquisition. This will look like this for 2D or 3D experiments:



Other SMIS features

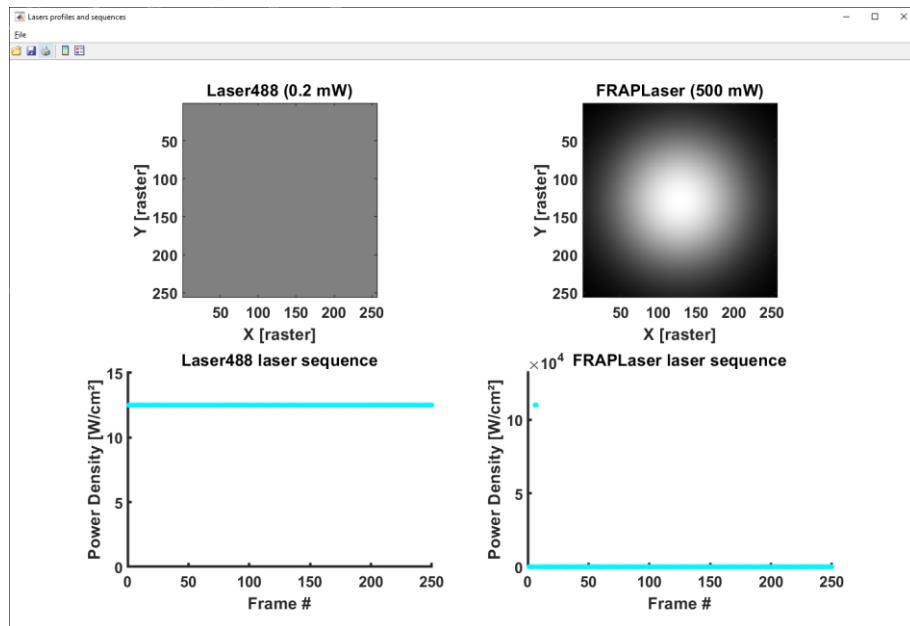
Monitoring what's going on

At the bottom of the main SMIS window, you can use the 3 red buttons to look at lasers, fluorophore spectra and point spread functions:



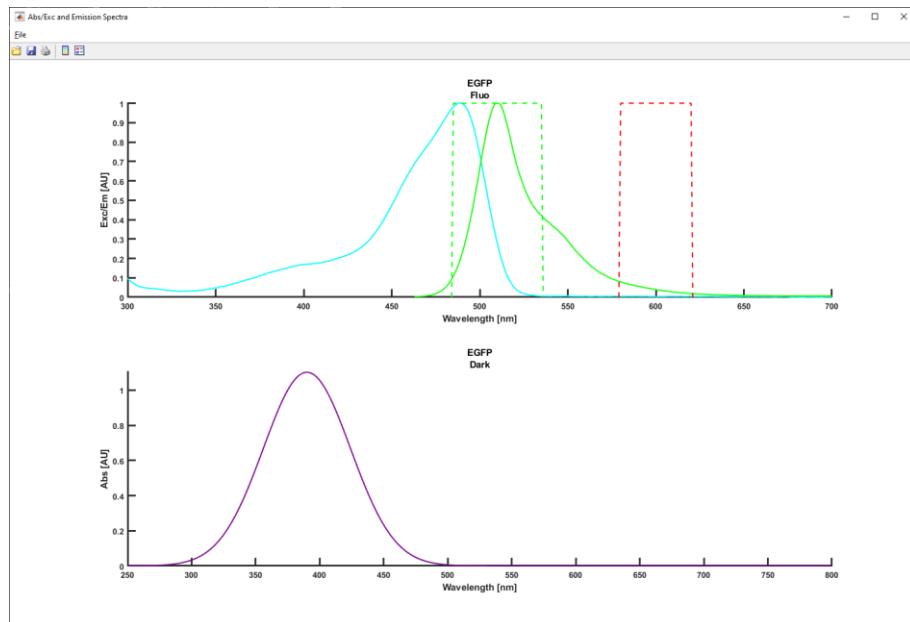
Lasers

Click the « *Show lasers* » button to show a summary of the defined lasers, as in the example below:



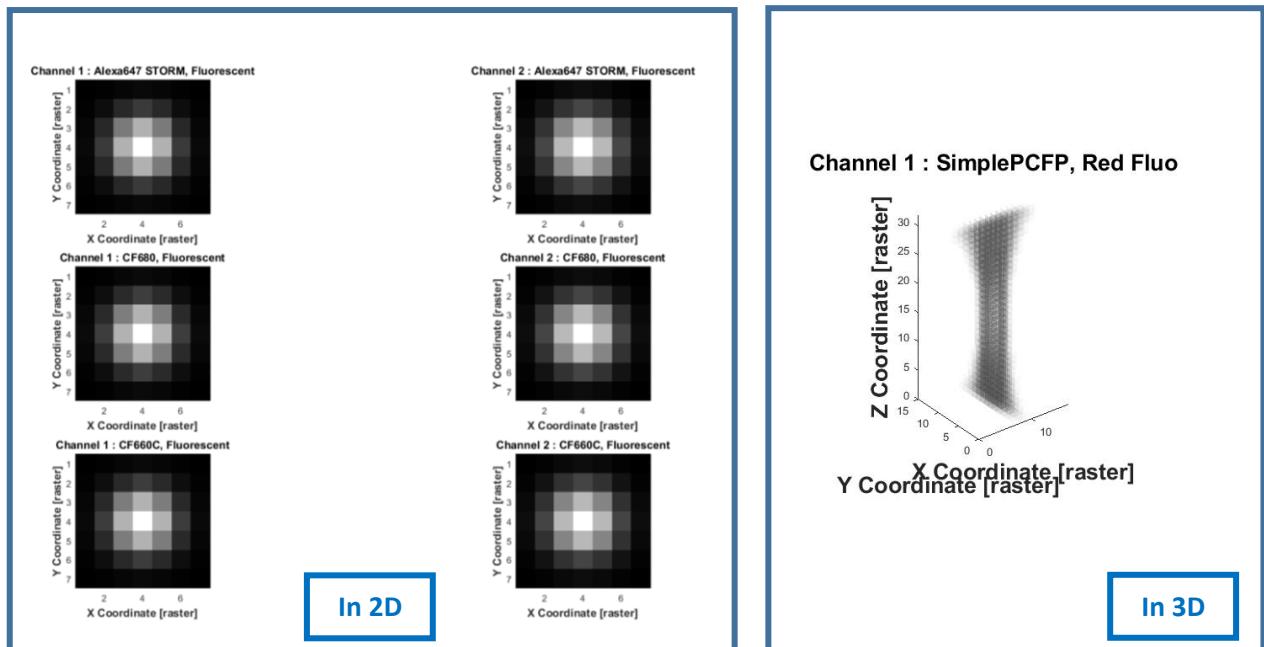
Fluorophore spectra

Click the « *Show fluorophore spectra* » button to show Absorption/Excitation and fluorescence emission spectra of the fluorophores, as in the example below:



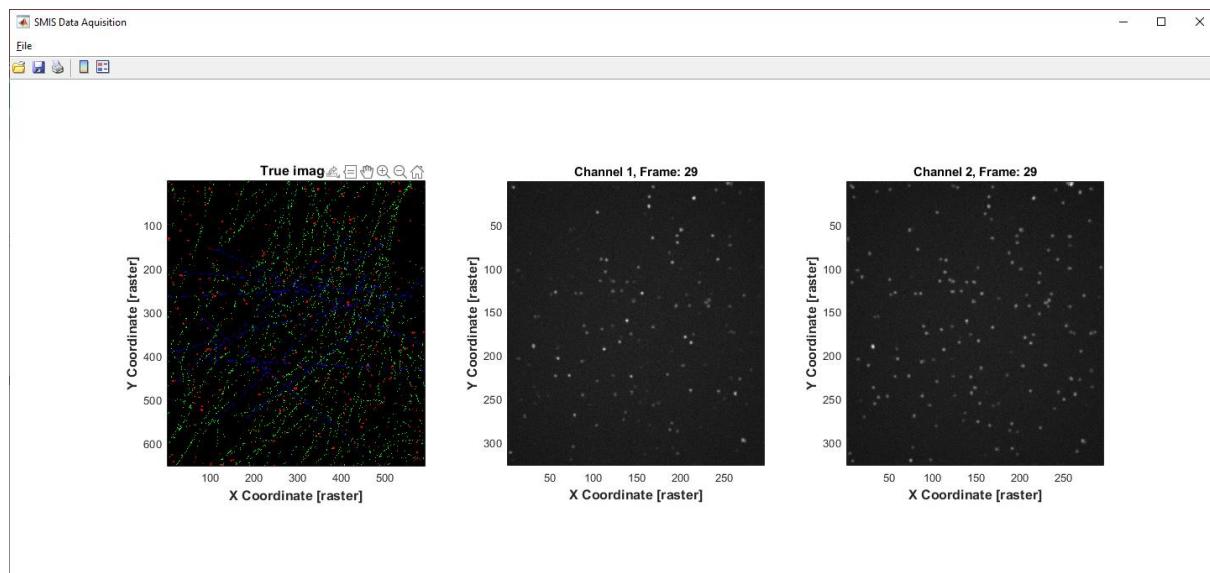
PSF

Click the « Show PSF » button to show the point spread function corresponding to each fluorophore in all their fluorescent states for each channel, as on the examples below (2D and 3D):



Monitoring data collection

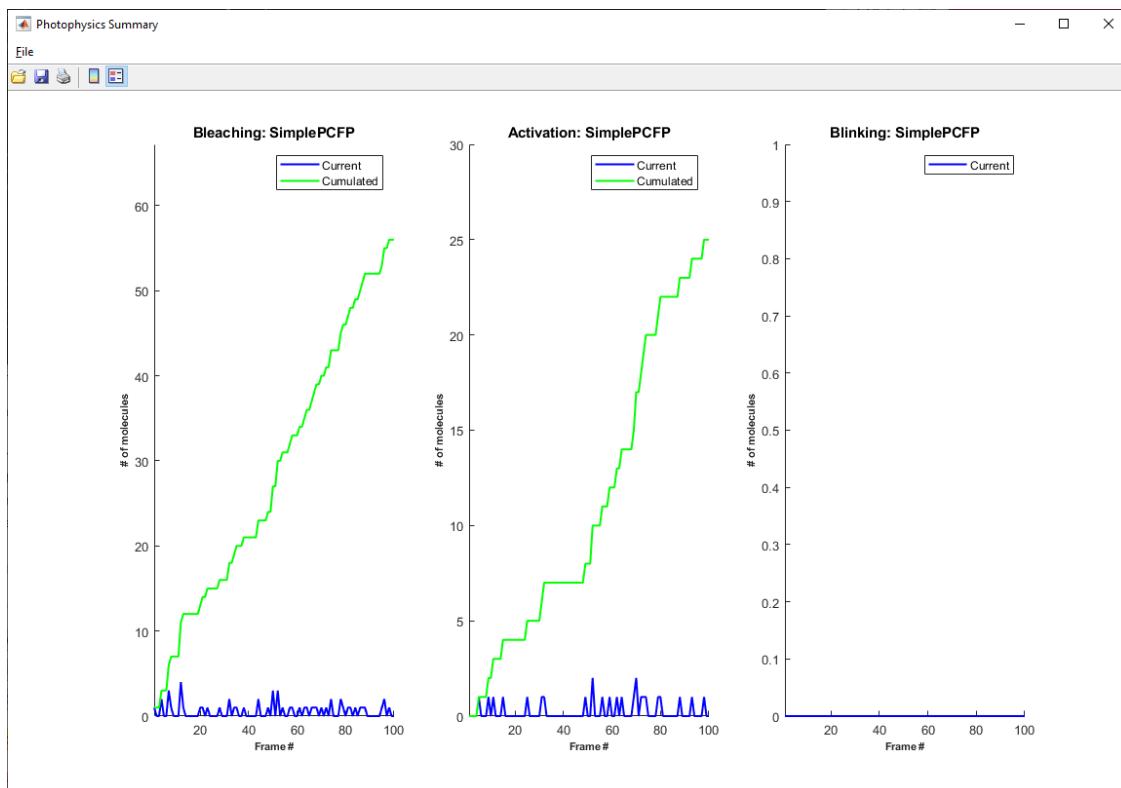
Set « Show that during acquisition » to on, to visualize the generated frames during simulation. You will see on the left the ground truth fluorophores decorating the virtual sample, and on the right the collected frames, either one channel or two channel.



Setting this option on significantly slows down SMIS, so it is advisable to only use this when setting up the simulation, not for the final run.

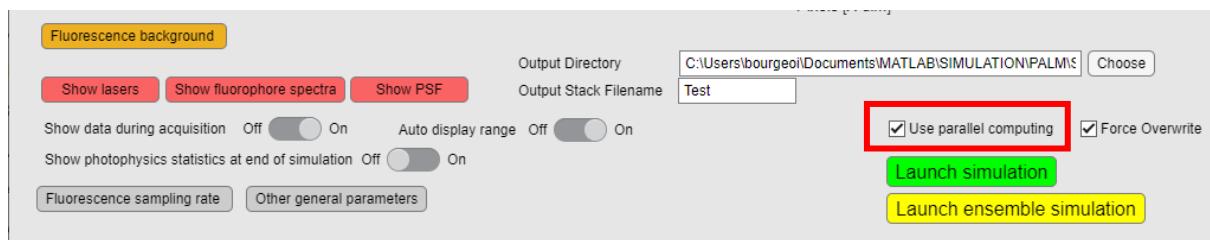
Monitoring photophysics

Set « *Show photophysics statistics at end of simulation* » to get a grasp of the photophysical states of the fluorophores along simulation.



Parallel computing

You can use parallel computing in SMIS by ticking « *Use parallel computing* »



Parallel computing is generally useful when you have long simulations to run with many molecules per fluorophore. Note that starting parallel computing takes time, so there will be a lag in starting SMIS simulations. Typically also the first few frames will be quite slow, and then there will be a very significant acceleration. The efficiency of parallel computing really depends on the type of simulation you are running, and is not guaranteed. The speed increase will not be proportional to the number of available CPUs, although the SMIS code has been designed to cope at best with the requirement of parallel computing. We typically observe a maximum speed gain of a factor of 4 to 5 in simulations adequate for parallel computing, ie with > 1000 molecules.

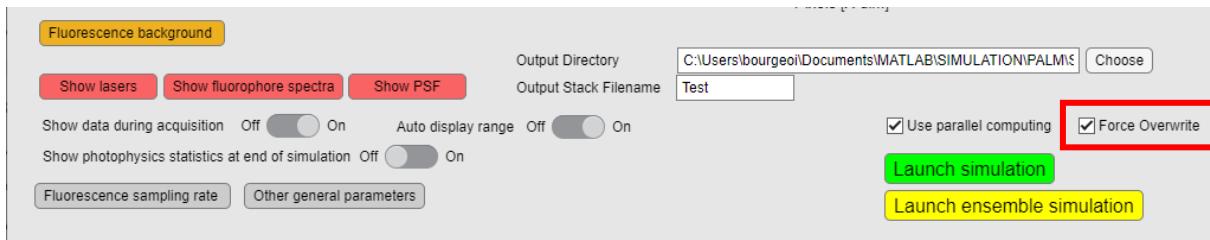


If you are running quick simulations with few molecules, there will be a strong time penalty in using parallel computing

As explained in the section “Other general parameters”, you can set the minimum number of molecules required to allow turning on parallel computing: below this number, even if parallel computing is set on, it will be turned off because it will be very inefficient. During a simulation starting with a high number of molecules, when the number of unbleached molecules will become below the entered minimum number, parallel computing will be automatically switched off.

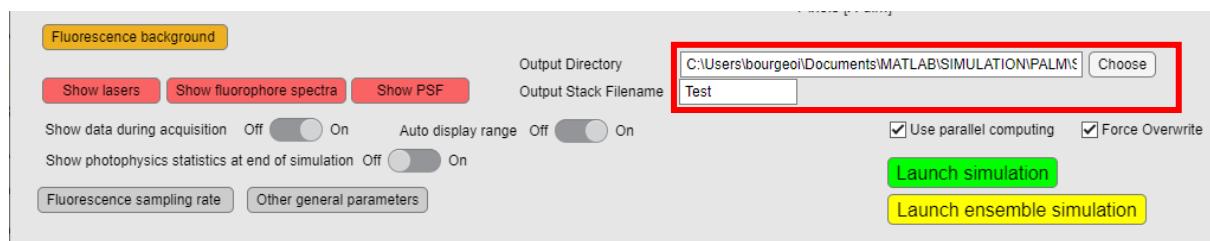
Force overwrite

You can force overwriting simulation data without confirmation by ticking « Force Overwrite »

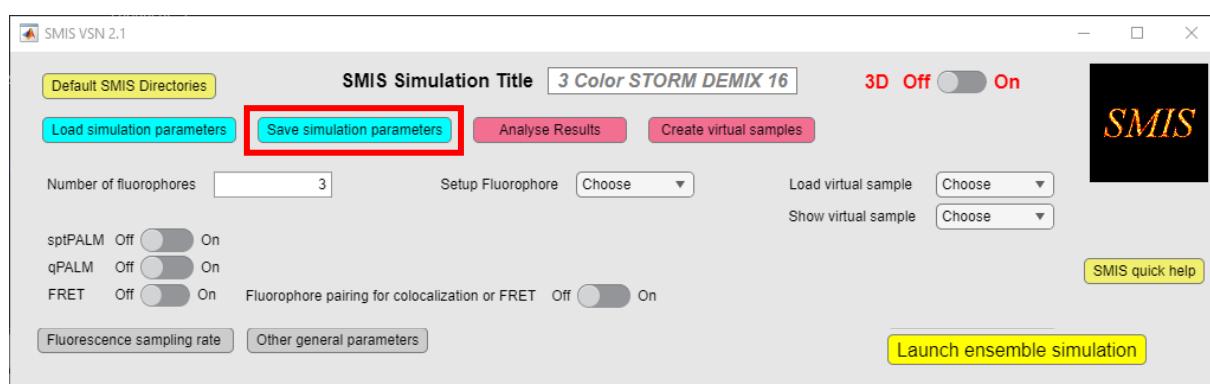


Launching simulations

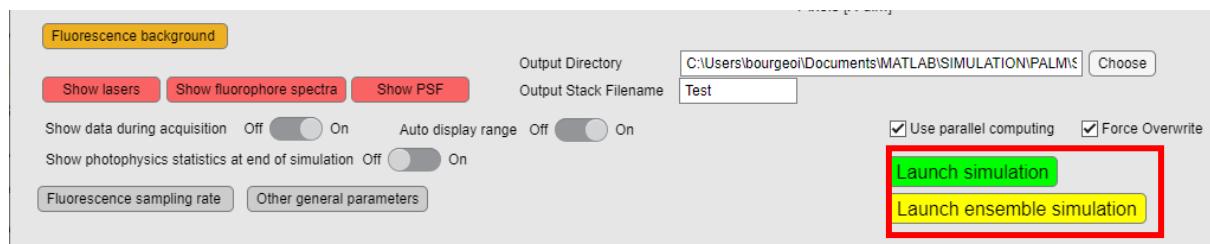
Before launching a SMIS simulation, define the location of the output data:



And maybe save again the simulation parameters:

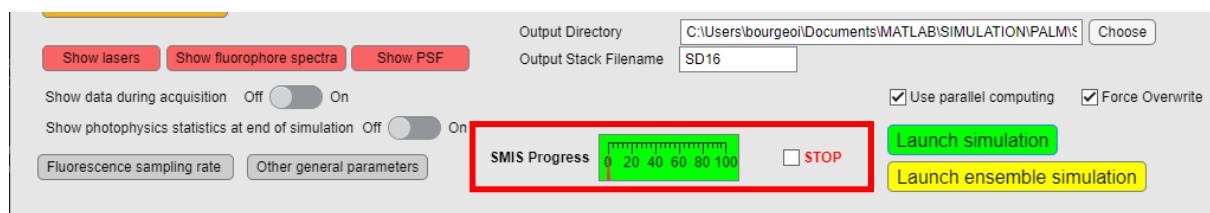


When you are finally ready to launch your simulation, you can do it either at the single molecule level or at the ensemble level.



Click on « *Launch simulation* » for single molecule simulation and click on « *Launch ensemble simulation* » for an ensemble level simulation. Ensemble level simulations will run a lot quicker, and are very useful to check parameters and notably the evolution of populations of photophysical states along the planned data collection scheme.

Once assimilation has been started, you can monitor its progression and stop it by clicking on the « *Stop* » button.



Stopping a simulation can lag for a number of frames depending on the case. Just be a little patient !

A lot of information on the progress is display on the SMIS monitoring window (Standalone SMIS) or on the Matlab command window.

```

Sélection SMIS_2.1
Elapsed time for this frame: 0.35423 seconds
Processing frame #: 127
Laser 1 (Laser642): Power: 56.8877 mW, Wavelength: 642 nm
Laser 2 (Laser405): Power: 0.0075 mW, Wavelength: 405 nm
# of bleached molecules in this frame (Alexa647_STORM): 0
# of activated molecules in this frame (Alexa647_STORM): 0
# of blinked molecules in this frame (Alexa647_STORM): 87
Cumulated # of bleached molecules (Alexa647_STORM): 13
Cumulated # of activated molecules (Alexa647_STORM): 100
# of bleached molecules in this frame (CF680): 0
# of activated molecules in this frame (CF680): 0
# of blinked molecules in this frame (CF680): 44
Cumulated # of bleached molecules (CF680): 6
Cumulated # of activated molecules (CF680): 50
# of bleached molecules in this frame (CF660C): 0
# of activated molecules in this frame (CF660C): 0
# of blinked molecules in this frame (CF660C): 39
Cumulated # of bleached molecules (CF660C): 11
Cumulated # of activated molecules (CF660C): 50
Elapsed time for this frame: 0.41315 seconds
Processing frame #: 128
Laser 1 (Laser642): Power: 56.8877 mW, Wavelength: 642 nm
Laser 2 (Laser405): Power: 0.0075 mW, Wavelength: 405 nm
# of bleached molecules in this frame (Alexa647_STORM): 0
# of activated molecules in this frame (Alexa647_STORM): 0
# of blinked molecules in this frame (Alexa647_STORM): 87
Cumulated # of bleached molecules (Alexa647_STORM): 13
Cumulated # of activated molecules (Alexa647_STORM): 100
# of bleached molecules in this frame (CF680): 0
# of activated molecules in this frame (CF680): 0

```

Output Data

SMIS will generate a series of output files. If your simulation is named “MySMIS”, the following files are created:

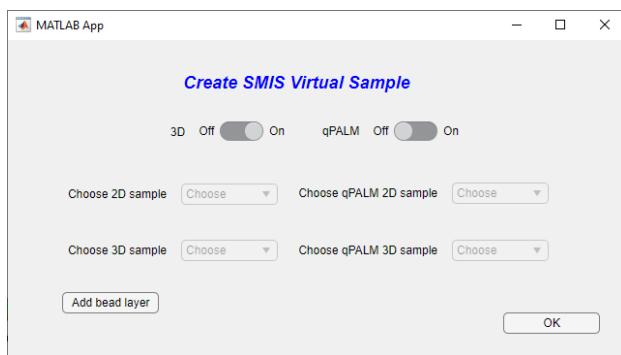
- 1/ Image stack (“MySMIS_ch1.tif”). If 2 acquisition channels are used with 2 virtual CCD cameras, a second image stack is output (“MySMIS_ch2.tif”).
- 2/ A diary file (“MySMIS_diary.txt”), describing the progress of the simulation.
- 3/ A Matlab file (“MySMIS.mat”), containing the ground truth information for the fluorophores (see Annexes).
- 4/ Projections of the image stacks (“MySMIS_ch1_dl.tif”) (and possibly “MySMIS_ch2_dl.tif”), providing diffraction limited images of the sample.
- 5/ Beam profile for each laser (“MySMIS_Mylasernames.tif”).
- 6/ View of the ground-truth labeled virtual sample (“MySMIS_true.tif”)

Creating virtual sample



Upon choosing « *Create virtual samples* » in the main SMIS window, the following window will appear:

Different types of virtual samples can be created, either for 2D or 3D SMIS simulations, and either for regular SMLM or for qPALM (see the chapter on virtual samples at the beginning of the SMIS manual). In 3D mode, you can add a “bead layer” to your sample, in case for example you want to simulate drift, and drift correction based on bead drift monitoring.



The number of possibilities offered by this SMIS tool is for now limited, but could be expanded in future versions of SMIS. Of course you can create more virtual samples using segmented images e.g. in Fiji or in Matlab.

Below 2 examples are given:



One important point to notice is that virtual samples are digital objects, so that you define dimensions in terms of pixel numbers, which are then converted into physical size through both the pixel size and the binning factor defined in SMIS. This is why the virtual sample dimensions is displayed on the right size, and change according to the chosen binning and pixel size. You will have to remember in SMIS simulations what binning and pixel size values correspond to which physical dimensions.

Note that you can also load an existing sample of the same type, to facilitate the creation of a new similar sample.

Analyzing results



Upon choosing « *Analyze Results* » in the main SMIS window, the following window will appear:



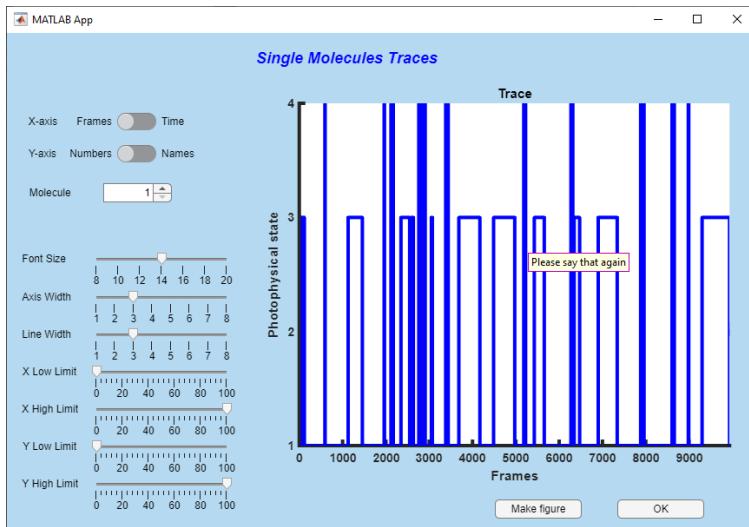
Analyzing single molecule data

1. You can choose between analysis of single molecule (*SM*) or ensemble data (*Ensemble*). The window shown above is for single molecule data.
2. Then load the SMIS simulation ground truth output data of your choice (.mat file)
3. If this is a multicolor simulation, choose here the fluorophore you want to look at.
4. You may want to only select molecules that have “completed their whole life”, i.e. bleached molecules.
5. Select the photophysical state, or other photophysical parameters you want to look at. The other parameters are: photon budget (total number of detected photons before bleaching), photons per localization (number of detected photons per localization), global on times (i.e. on times as seen by the detector), global off times (i.e. off times as seen by the detector). For photon budget and photons per localization, if this is a multichannel experiment you can choose the channel in which to look at.

You can fit the histograms using monoexponential, bi-exponential or stretched-exponential models. Click on *fit* and select the type of fit, and then click again on *show* to display the fit. Fitting parameters are then displayed below the table, and the fit are overlaid on the histograms. You may want to remove the 1st point of the histogram when performing fits. For global on and off times you may choose a “blinking gap” to link *on* times separated by a few frames. You may also compute the histograms only from a defined frame number, in case for example you have a bleaching phase at the beginning of the data acquisition.

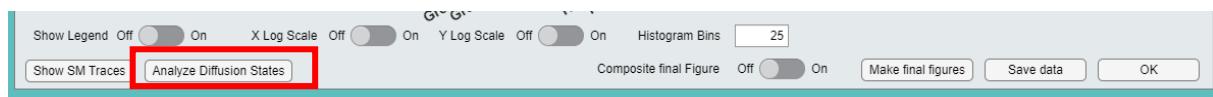
Use this parameters to adjust the plots the way you like.

6. You can play with data presentation using these various tools.
7. You can output a nice final figure. In the table select *Plot final* for those states or photophysical parameters that you want to plot, and then click on *Make final figure*. You can create individual output figures, or a single composite figure using the *Composite final figure* switch. In the latter case you may have to expand the output figure to nicely see the plots.
8. You can save the analyzed data for further inspection within Matlab.
9. You can look at single molecule traces individually. This will open a new window:

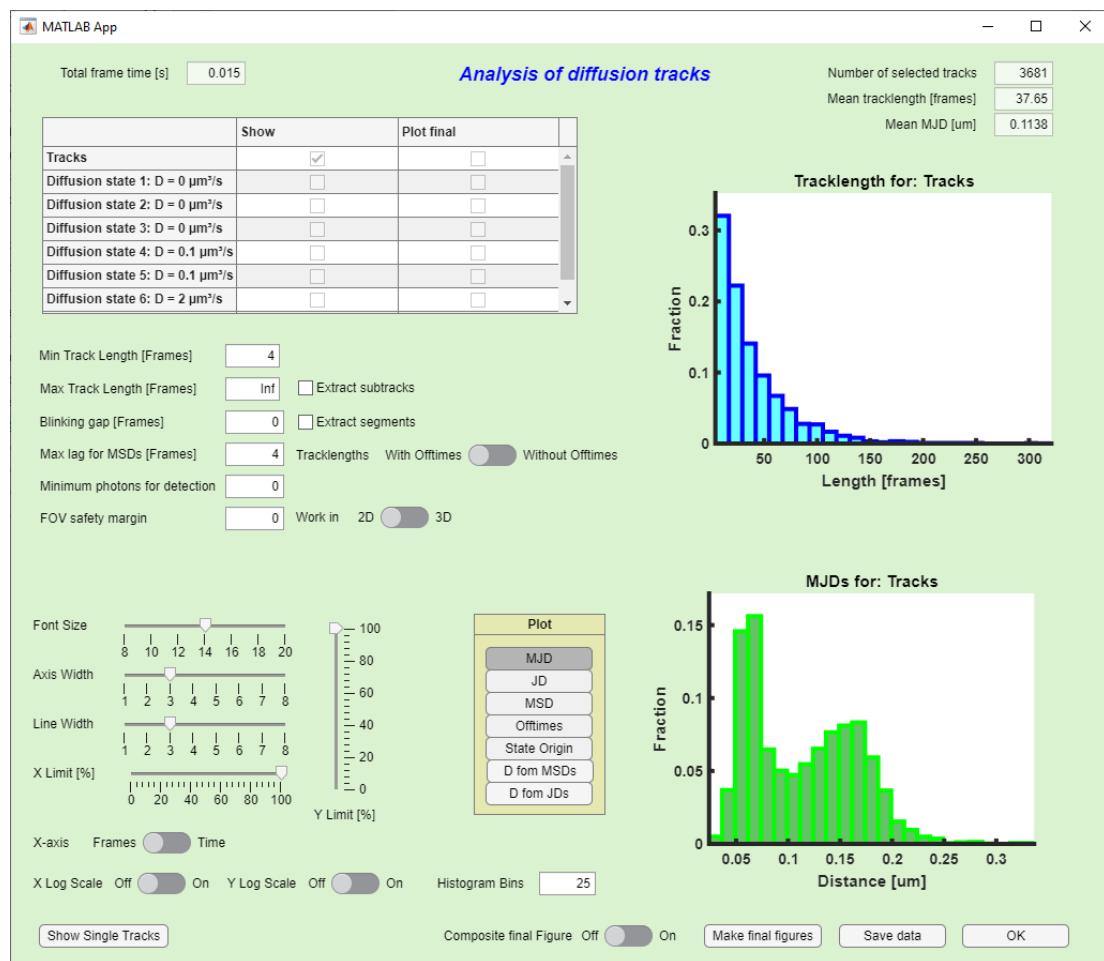


Enjoy looking at your traces, you can create a figure if needed.

10. If you are looking at SPT data, you will have the option to look at the tracks:



This will open the following window. Note that a number of calculations will be performed to calculate tracks, segments, mean square displacements, jump distances etc ...



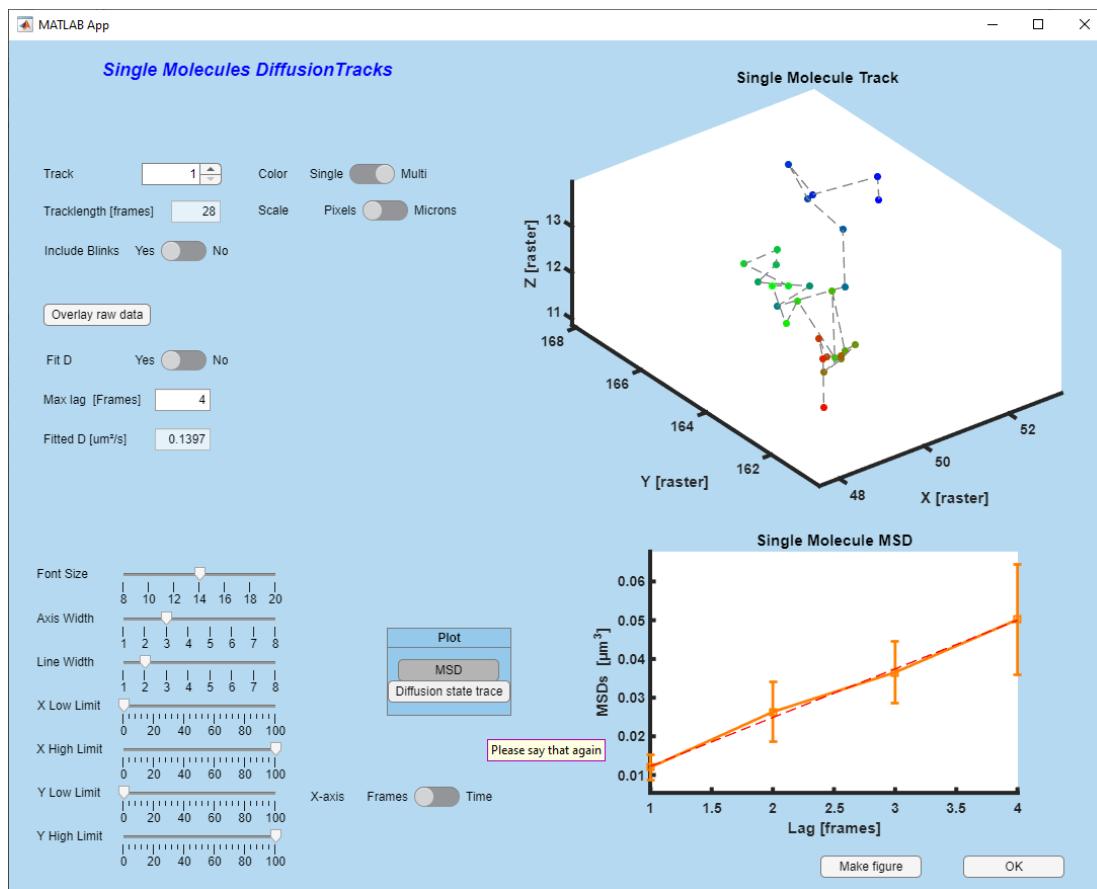
You can do quite many things here: look at entire tracks, subtracks (pieces of tracks that are separated by long off times) or segments (pieces of tracks that correspond to a certain photophysical state). You can choose parameters to refine the ensemble of tracks or subtracks you want to look at. For each of those, you can plot jump distances, mean square displacements, and calculate diffusion coefficients. You can fit diffusion coefficient histograms with various models.



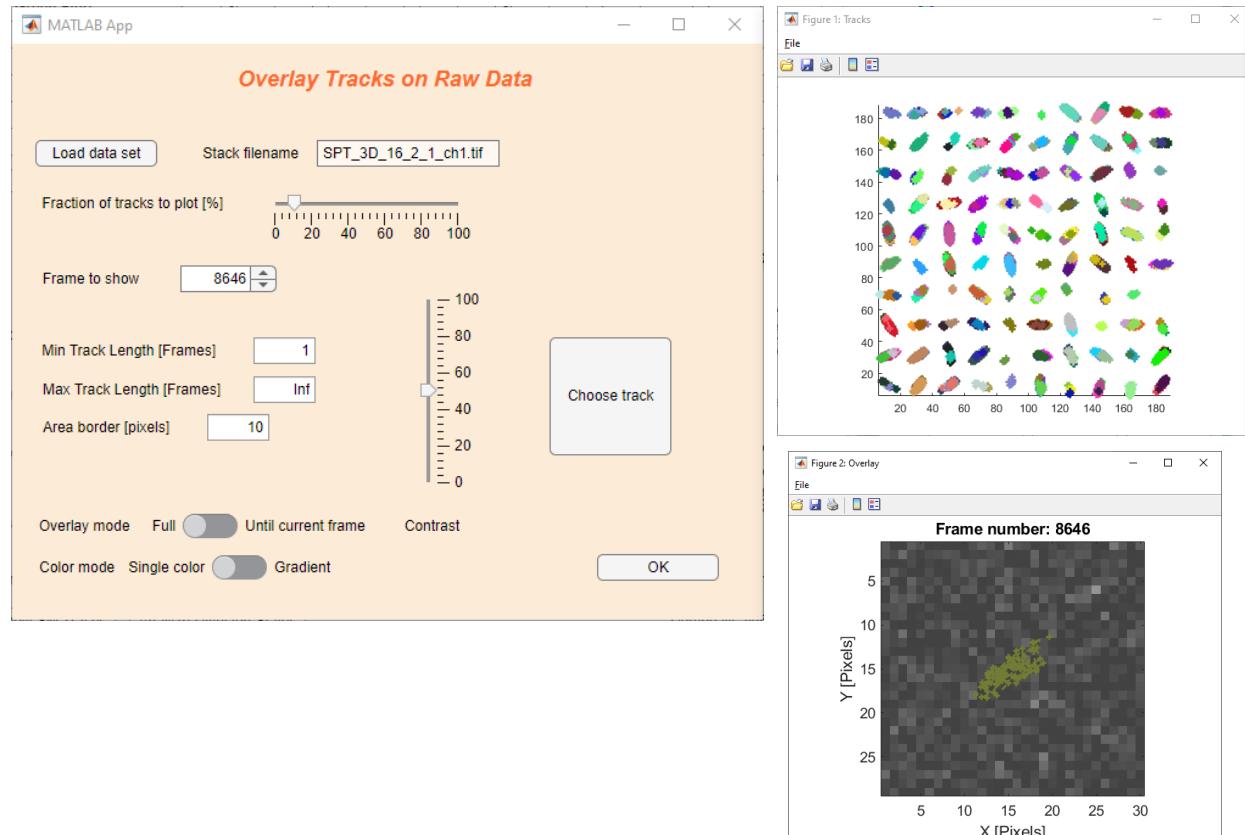
Calculations can take quite long ... be patient !

A detailed description of all what you can do is left for the next version of this manual ... But look at the online help by moving the mouse over the different buttons.

Finally you can also look at individual tracks, in 2D or 3D:

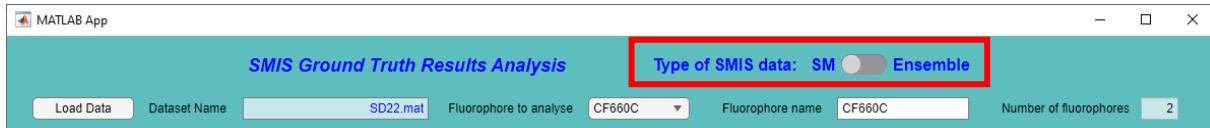


And you can even overlay tracks on raw data:

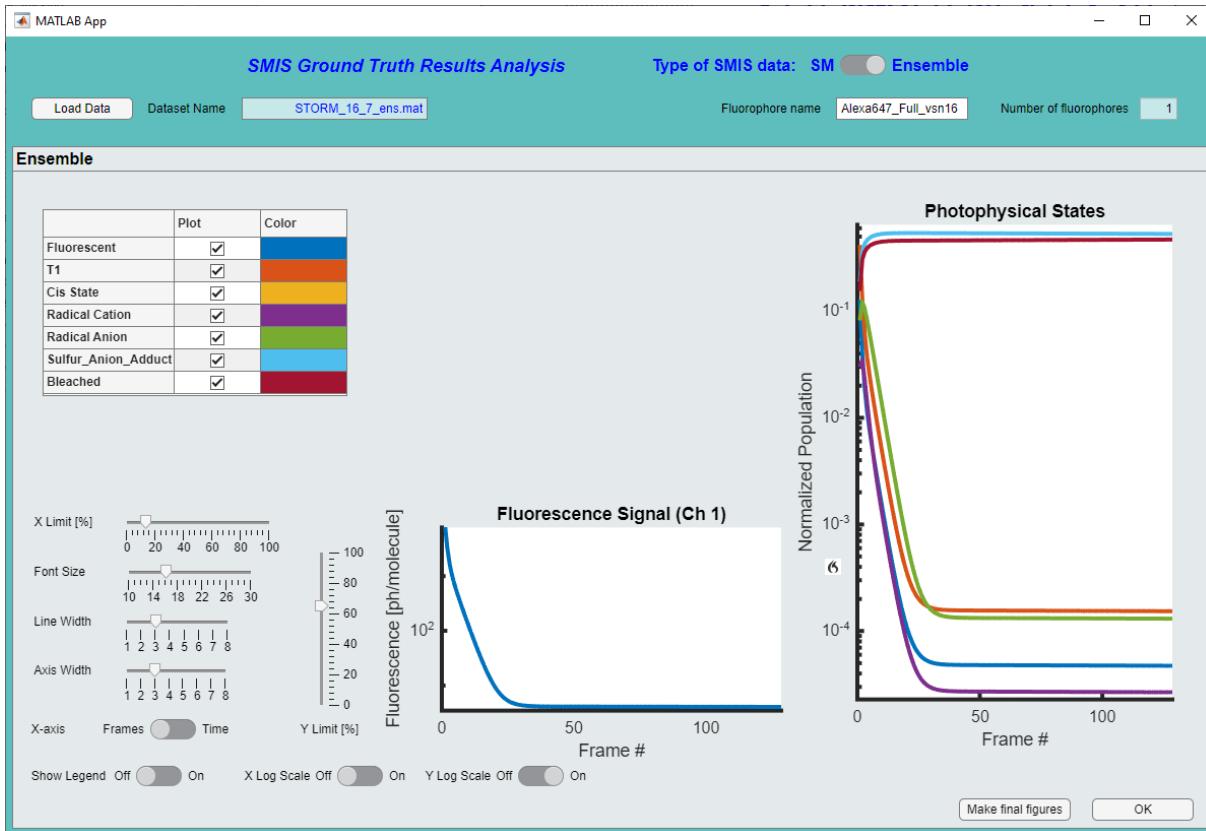


Analyzing ensemble data

If you want to analyze ensemble data, switch to the ensemble type of SMIS data.



The following window will appear:



Enjoy looking at data, you can select the fluorophores you want to look at, set different types of scaling, etc. and output a nice figure if needed.

Annex 1: Content of the *sms* Matlab structure

All ground truth single molecule data are found in a Matlab structure array called '*sms*'. The sub-structure *sms*(*i*).*sm* will contain the data for fluorophore # *i*. The *sm* structure looks like:

```

sm(1:sm_par(i).n_mol)=struct(
    'x',0, ... % xyz position in current frame (may vary with drift) [in pixels] in high-resolution
image
    'y',0, ...
    'z',0, ...
    'x_track',[], ... % list of xyz positions along the diffusion track [in pixels] in high-
resolution image
    'y_track',[], ...
    'z_track',[], ...
    'sub_x',[], ... % list of xyz positions along the sub diffusion track [in pixels] if
use_diffuse_psf option is used
    'sub_y',[], ...
    'sub_z',[], ...
    'v_x',[], ... % xyz velocity in current frame (may vary with drift) [in pixels] in high-
resolution image if directed diffusion is used
    'v_y',[], ...
    'v_z',[], ...
    'c_sp',[], ... % # of the subpattern onto which the sm is placed along diffusion
    'n_sp',[], ... % # of the subpattern onto which the sm is placed along diffusion when/if
transition occurs
    'id',0,... % id # of each molecule
    'theta',sm_par(i).theta_fixed,'phi',sm_par(i).phi_fixed,... % theta & phi polar angles. Only
used if anisotropy=1
    'state', sm_par(i).initial_state, ... % Current photophysical state
    'state_id', sm_par(i).state_ids(sm_par(i).initial_state), ... % Id of current photophysical
state (several states can have the same id if they interconvert rapidly and cannot be distinguished
    'state_trace', [], ... % Shows photophysical state evolution along current frame
    'fluo_trace', [], ... % Shows when the molecule is in fluorescent state along current frame
    'tot_state_trace', [0;sm_par(i).initial_state], ... % Shows photophysical state evolution along
all frames until bleaching
    'tot_fluo_trace', [], ... % Shows when the molecule is in fluorescent state along all frames
until bleaching
    'Ns', zeros(int8(im_par.addtime>0)+1,sm_par(i).n_states), ... % # of absorbed photons per
sampling time in each starting photoactive state for addtime and frametime
    'sampling_rate', nan(1,int8(im_par.addtime>0)+1), ... % local sampling rate for addtime and
frametime (might differ depending on sm position)
    'n_abs', zeros(1, sm_par(i).n_states), ... % # of absorbed photons (current frame)
    'n_em',zeros(1, sm_par(i).n_fluorescent_states), ... % # emitted photons for each fluorescent
state in current frame
    'tot_n_em',zeros(1, sm_par(i).n_fluorescent_states), ... % total # of emitted photons for each
fluorescent state
    'n_phot_ch1', zeros(1, sm_par(i).n_fluorescent_states), ... % # emitted photons in ch1 for each
fluorescent state (current frame)
    'n_phot_ch2', zeros(1, sm_par(i).n_fluorescent_states), ... % # emitted photons in ch2 for each
fluorescent state (current frame)
    'tot_n_phot_ch1', zeros(1, sm_par(i).n_fluorescent_states), ... % total # of emitted photons in
ch1 for each fluorescent state (all frames)
    'tot_n_phot_ch2', zeros(1, sm_par(i).n_fluorescent_states), ... % total # of emitted photons in
ch2 for each fluorescent state(all frames)
    'n_phot_det_ch1', [], ... % total # of detected photons in ch1 (current frames)
    'n_phot_det_ch2', [], ... % total # of detected photons in ch2 (current frames)
    'tot_phot_det_ch1', 0, ... % total # of detected photons in ch1 (all frames)
    'tot_phot_det_ch2', 0, ... % total # of detected photons in ch2 (all frames)
    'frames_on_ch1', [], ... % frames where the molecule is on for channel 1 (ie, emits photons)
    'frames_on_ch2', [], ... % frames where the molecule is on for channel 2 (ie, emits photons)
    'bleached', 0, ... % 0: not bleached; 1: bleached
    'activated', 0, ... % 0: not activated; 1: activated
    'blinked', 0, ... % 0: not switched; 1: switched
    't_on', zeros(1, sm_par(i).n_fluorescent_states), ... % time the sm is in fluorescent states
(current frame)
    'fr_t_on', [], ... % Fraction of times the sm is in fluorescent states (current frame) (for
diffuse PSF calculations)
    't_off', zeros(1, sm_par(i).n_dark_states), ... % time the sm is in dark states (current frame)
    'diff_state', [], ... % Current Diffusion state
    'n_diff_state', [], ... % New Diffusion state when/if transition occurs
    'diff_state_trace', [;], ... % History of diffusion state
)

```

```

'em_spectrum',[;], ... % Emission spectrum (current frame)
'given_fret_photons',[;], ... % Transferred # of photons for each fluorescent state when molecule
is donor
'received_fret_photons',[;], ... % Transferred # of photons for each photoactive state when
molecule is acceptor
'fret_eff', 0, ... % FRET efficiency to acceptor (if donor)
'matched',0, ... % index of partner molecule (if dyes_pair_on is set)
'lx',[;], ... % x linkage error [raster unit]
'ly',[;], ... % y linkage error [raster unit]
'lz',[;], ... % z linkage error [raster unit]
'le_set',[;] ... % 1 if linkage error is set for current frame
);

```

Annex 2: Content of the *sm_par* Matlab structure

For each fluorophore, some general parameters are found in the “*sm_par*” Matlab structure array. The sub-structure *sm_par*(*i*) will contain the parameters for fluorophore # *i*. The structure looks like:

```

sm_par(1:n_fluorophores)=struct(...
    'fluorophore_name', [], ...
    'n_mol', [], ...
    'n_mol_eff', [], ...
    'maturation_level', [], ...
    'pH_sensitivity', 0, ...
    'pH', 7.5, ...
    'spectral_data', [], ...
    'n_states', [], ...
    'state_ids', [], ...
    'initial_state', [], ...
    'initial_fluo_state', [], ...
    'fluorescent_states', [], ...
    'associated_dark_states', [], ... % In case fluorescent states are in rapid equilibrium
with dark states
    'fluorescent_fraction', [], ... % In such case, fraction of the fluorescent states
    'converted_state', [], ...
    'bleached_states', [], ...
    'trans_k', [], ...
    'sampling_rate', sampling_rate, ... % sampling rate for photophysics during frametime
and addtime [s-1]
    'start_sampling_rate', nan(1,int8(numel(sampling_rate)==2)+1), ... % starting sampling
rate for photophysics during frametime and addtime [s-1] after optimization
    'trans_q', [], ...
    'photoactive_states', [], ...
    'quantum_yield', [], ...
    'fluorogenic', [], ...
    'fluorogenicity', [], ...
    'psf_par_ch1', [], ...
    'psf_par_ch2', [], ...
    'filter_profiles', [], ...
    'D', [], ...
    'D_ex_rates', [], ...
    'D_confined', [], ...
    'D_rate_matrix', [], ...
    'V', [], ...
    'persistence_length', [], ...
    'dispersion_selectivity', 3, ... % power factor telling how much a sm will prefer
choosing wide channels relative to narrow ones
    'margin_factor', 1e-5, ... % Margin factor to avoid velocity vector ending up being
zero when molecule close to image border
    'V_circle', [], ...
    'V_init_dir', [], ...
    'w_patterns', struct('w', []), ...
    'anisotropy', [], ...
    'dipole_orientation', [], ...

```

```



```

Annex 3: Content of the *im_par* Matlab structure

Finally the SMIS virtual microscope is described by some general parameters found in the “*im_par*” Matlab structure. The structure looks like:

```

%% Define general parameters for the images
im_par=struct(
    'n',[], ... % X size of pattern
    'm',[],... % Y size of pattern
    'nz',[], ... % Z size of pattern
    'raster',raster,...,
    'binning', binning, ...
    'd1d2_dist',d1d2_dist, ...
    'd1d2_dist_sig', d1d2_dist_sig, ...
    'd2_constrained', d2_constrained, ...
    'obj', [], ... % objective parameters
    'det', [], ... % detector parameters
    'bg', [], ... % Background parameters
    'filters', [], ... % Filters parameters
    'n_images', n_images, ...
    'current_frame', 0, ...
    'frametime', frametime, ...
    'addtime', addtime, ...
    'during_frametime', [], ...
    'add_drift', add_drift, ...
    'add_diffusion', add_diffusion, ...
    'rbox_ch1', 0, ...
    'rbox_ch2', 0, ...
    'distribute_in_clusters',distribute_in_clusters, ...
    'simul_3D',simul_3D, ...
    'psf_mode',psf_mode, ...
    'psf_astigmatism_x',psf_astigmatism_x, ...
    'psf_astigmatism_y',psf_astigmatism_y, ...
    'psf_n_zslices', psf_n_zslices, ...
    'move_non_activated_molecules', move_non_activated_molecules, ...
    'use_diffuse_psf', use_diffuse_psf, ...
    'prevent_diffusion_out_FOV', prevent_diffusion_out_FOV, ...
    'diffuse_psf_radius', diffuse_psf_radius, ...
    'sample_zcenter', sample_zcenter, ...
    'depth_of_focus', depth_of_focus, ...
    'stochastic_spectrum_off',stochastic_spectrum_off, ...
    'two_channel', two_channel, ...
    'two_channel_defocus', two_channel_defocus, ...
    'two_channel_deform', two_channel_deform, ...
    'single_CCD', single_CCD, ...
);

```

```
'mol_density', 0, ...
'fret_on', fret_on, ...
'fluorophore_pairing_on', fluorophore_pairing_on, ...
'apply_poisson_stat_for_lasers', apply_poisson_stat_for_lasers, ...
'optimize_sampling_rate', optimize_sampling_rate, ...
'minimum_oversampling', minimum_oversampling, ...
'min_sampling_points', min_sampling_points, ...
'ex_rates_min_oversampling', ex_rates_min_oversampling, ...
'rng_state', rng_state, ...
'use_parallel_computing', use_parallel_computing, ...
'min_n_mol_pct', min_n_mol_pct, ...
'poolobj', [], ...
'debug', debug);
```