Manual

A manual for maximum likelihood analysis of 2- and 3-color photon trajectories of fluorescence bursts collected from freely diffusion molecules using a confocal microscope (linear models, Section 4) and customization guides for arbitrary models (Section 5). BurstML and colorML analyses using Mathematica and examples can be found at **\MathematicaAnalysis**. Explanations and instructions for the Mathematica analysis codes can be found in the Mathematica notebook, **burstMLcolorML.nb**. The two example files (**burstsSampleEqual**, **burstsSampleUnEqual**) are provided for convenience because it takes time to generate in the notebook.

Contents

1. Requirements

MATLAB (≥ 2019b) The analysis program has been developed and tested on 2019a and 2023a.

In order to use pre-compiled MEX (https://www.mathworks.com/help/matlab/matlab_external/introducing-mex-files.html) analysis functions, you need MS windows (x64) and Visual C++2022 Community (https://visualstudio.microsoft.com/downloads/). Otherwise, see Section 5 to compile the MEX functions on your system.

2. Installation

Simply unzip to extract files and add the file location to your MATLAB path (typically, it takes less than a minute). For the examples in the script files, copy functions in 'private' directory to one of your MATLAB path.

3. Input files

The photon trajectories need to be formatted in a specific way.

3.1. Preprocessing of photon trajectories: burst selection

The raw data is records of individual photon's arrival time and color and additional delay time from a laser pulse when pulsed laser excitation is used. Significant bursts are identified for further analysis using two criteria: interphoton time threshold τ_{th} and photon number threshold N_{th}. All photons with interphoton time $\tau < \tau_{\mathrm{th}}$ are combined into a burst and a burst with number of photon $N \geq N_{\mathrm{th}}$ are analyzed using burstML. Burst selection script is included in a MATLAB .m file in each analysis example.

3.2. Photon trajectories

A preprocessed photon trajectory data consists of a 5-column matrix. The first column elements are burst indices determined in Section 3.1. The second column element is spared for bin indices of photons, which is not used in burstML analysis. The third column elements are photon arrival times in 100 ns unit. The fourth column corresponds to delay time from the laser pulse trigger signal, which is used for identification of photons by donor and acceptor 1 laser excitation in pulsed interleaved excitation (PIE). This is not used in the analysis of the data collected by continuous-wave mode laser excitation. The fifth column elements correspond to photon color or detection channel. In the 2-color analysis, acceptor and donor photon indices are 1 and 2, respectively. In the 3-color analysis with CW excitation, acceptor 1, acceptor 2 and donor photon indices are 1,2, and 3, respectively. In the 3-color PIE analysis, there are additional indices of acceptor 1 and acceptor 2 photons by acceptor 1 excitation, which are 4 and 5, respectively. Photon trajectories of all bursts are contained in a single matrix.

3.3 Photon trajectory segment index

Photon trajectory segment (i.e., burst) indices locate the start and end photons of individual bursts in the photon trajectory matrix. For example, the range of the i-th photon segment in the photon trajectory matrix is V(i)+1 - V(i+1) (index starts from 1 in MATLAB).

4. Parameter Optimization

4.1. Static 2 components (2 color CW): mlhDiffNbkg_MT, mlhipt_MT

Preprocessing for burst selection and analysis of simulated trajectories are contained in **analysisStatic2comp.m**. The first section lists parameters for simulated diffusion photon trajectories. The second section (burst selection) is for burst selection and characteristic plots. The third section (burstML analysis) contains burstML1 and iptML analyses (switch variable isiptML). There are two types of photons: donor (D) and acceptor (A). Both **mlhDiffNbkg_MT** and **mlhipt_MT** work for *N* component model.

<u>Relevant literatures</u>:
Gopich IV, Kim J-Y, Chung HS, "Analysis of photon trajectories from diffusing single molecules" J. Chem. Phys. 159, 024119 (2023). https://doi.org/10.1063/5.0153114
Gopich IV, Chung HS, "Unraveling Burst Selection Bias in Single-Molecule FRET of Species with Unequal Brightness and Diffusivity" J. Phys. Chem. B 128, 5576 (2024). https://doi.org/10.1021/acs.jpcb.4c01178

4.2. 2-state folding (2 color CW): mlhDiffNTRbkg_MT

Preprocessing for burst selection and analysis of simulated trajectories are contained in **analysisKinetic2component.m**. The first section lists parameters for simulated diffusion photon trajectories. The second section (burst selection) is for burst selection and characteristic plots. The third section (burstML analysis) contains burstML1 analysis. There are two types of photons: donor (D) and acceptor (A). **mlhDiffNTRbkg_MT** works for *N*-state linear kinetic model ($S_1 \leftrightarrow S_2 \leftrightarrow \ldots \leftrightarrow S_N$).

4.3. 2-state folding (3 color CW): mlhDiffNTRbkg_MT, mlhDiffNTR3cDA1CWbkg_MT

Preprocessing for burst selection and analysis of simulated trajectories are contained in **analysisKinetic3colorFolding.m**. The first section lists parameters for simulated diffusion photon trajectories. The second section (burst selection CW) is for burst selection and characteristic plots. The third section (burstML analysis CW) contains burstML1 analysis. There are three types of photons: donor (D), acceptor 1 (A1) and acceptor 2 (A2). Two types of analyses are available (switch variable trjcolortype). The first analyzes bursts of molecules both with three active dyes (3c or DA1A2) and with two dyes (D and A1, DA1). The second analyzes only bursts produced by molecules with three active dyes (3c or DA1DA2). Both **mlhDiffNTRbkg_MT** and **mlhDiffNTR3cDA1CWbkg_MT** work for *N*-state linear kinetic model ($S_1 \leftrightarrow S_2 \leftrightarrow \ldots \leftrightarrow S_N$).

4.4. 2-state folding (3 color PIE/ALEX): mlhDiffNTR3cALEXbkg_MT, mlhDiffNTR3cDA1ALEXbkg_MT

Preprocessing for burst selection and analysis of simulated trajectories are contained in **analysisKinetic3colorFolding.m** as in the case of Section 4.3. This method analyzes photon trajectories produced by pulsed interleaved excitation (PIE), also called alternating laser excitation (ALEX) when excited by CW lasers. The first section lists parameters for simulated diffusion photon trajectories. The fourth section (burst selection ALEX) is for burst selection and characteristic plots. The fifth section (burstML analysis ALEX) contains burstML1 analysis. There are five types of photons: donor (D), acceptor 1 (A1) and acceptor 2 (A2) by donor excitation (Dex) and A1 and A2 by A1 excitation (Aex). Donor photons by Aex is ignored, so should be deleted prior to preprocessing. Two types of analyses are available (switch variable trjcolortype). The first analyzes bursts of molecules both with three active dyes (3c or DA1A2) and with two dyes (D and A1, DA1). The second analyzes only bursts produced by molecules with three active dyes (3c or DA1DA2). Both **mlhDiffNTR3cALEXbkg _MT** and **mlhDiffNTR3cDA1ALEXbkg_MT** work for *N*-state linear kinetic model ($S_1 \leftrightarrow S_2 \leftrightarrow \ldots \leftrightarrow S_N$).

4.5. 2-state binding (3 color CW): mlhDiffNTR3cDA1CWBindbkg_MT

Preprocessing for burst selection and analysis of simulated trajectories are contained in **analysisKinetic3colorBinding.m**. There are two molecular states (bound and unbound), but due to incomplete labeling, there are two bound states: three-color (3c or DA1A2) and DA1. Therefore, 3-color bursts and DA1 bursts should be analyzed together using a linear three state model B_3c ↔ U ↔ B_DA1. The first section of the code lists parameters for simulated diffusion photon trajectories. The second section (burst selection) is for burst selection and characteristic plots. The third section (burstML analysis) contains burstML1 analysis. There are three types of photons: donor (D), acceptor 1 (A1) and acceptor 2 (A2).

5. Custom models

To implement arbitrary model, you need to build your own function. For this purpose, you need additional requirements:

GSL library	For parameter optimization and linear algebra.
MATLAB library	To call C/C++ functions in MATLAB.
Visual studio 2022	Or any C/C++ compiler compatible with MATLAB.

5.1. Requirements

5.1.1. GSL library

The current version of the analysis functions was built with GSL version 2.4. You can download GSL from https://www.gnu.org/software/gsl/. For Windows OS, you can use a pre-built version for Visual Studio (https://www.bruot.org/hp/libraries/) or you can build GSL using Visual Studio https://github.com/BrianGladman/gsl.git.

5.1.2. MATLAB library

MATLAB library for MEX can be found at YOUR MATLAB PATHWAY /extern/lib. General C++ MEX Application guide can be found at https://www.mathworks.com/help/matlab/cpp-mex-file-applications.html

5.1.3 Visual Studio 2022

The current version of the analysis functions was built on Visual Studio 2022 (https://visualstudio.microsoft.com/downloads/).

5.2 Customization

Unzip the project file. You can find the source codes in "src" directory and MATLAB wrapper functions in "mWrapper" directory.

5.2.1. Visual Studio 2022

Modify the library paths in "x64MexPropertySheet.props". There are two paths to be modified:
C:\Program Files\MATLAB\R2019b This is your MATLAB2019 path.
C:\gsl\lib This is your GSL library path.

Replace these paths to your MATLAB and GSL installation paths.

5.2.2 Step-by-step guide for the installation and the path setting

1. Install MATLAB 2019 (for other MATLAB versions, please check MEX application documents for the corresponding version).
2. Install Visual studio 2022 community version from https://visualstudio.microsoft.com/downloads/
3. Install GSL library for Visual Studio 2022 from https://www.bruot.org/hp/libraries/
5. Modify paths in "x64MexPropertySheet.props" as described above.

5.2.3 How to introduce a new model

To introduce your own model, you need to check the following functions at least.

analysisThread This is the function for likelihood evaluation. Here you need to define your rate matrix and the way you evaluate the overall likelihood (including photon colors and number of states).

input from matlab This function assign MATLAB input double arrays into C/C++ variables. Also, there are some pre-defined constants for certain models.
init mlh vars This function initializes variables. Depending on a model vectors and matrices dimension need to be adjusted.

5.2.4 MATLAB version dependencies in Visual Studio

For MATLAB 2019 (and maybe newer versions), MW_NEEDS_VERSION_H needs to be defined in the headers for Visual Studio. And MEXFUNCTION_LINKAGE needs to be used to define the gateway function. In older versions, you can use _declspec(dllexport) to define the gateway function.