multipletau_cor_tttr Documentation

Release 0.1

Anders Barth

November 10, 2016

Contents

Contents:

CHAPTER 1

multipletau_cor_tttr Tutorial

1.1 Installing multipletau_cor_tttr

multipletau_cor_tttr is available as a module on PyPI, the Python Package Index.

If you have python installed, you can download *multipletau_cor_tttr* using:

```
$ pip install multipletau_cor_tttr
```

1.2 Using multipletau_cor_tttr

Import the module into you active python session or python script:

```
import multpletau_cor_tttr
```

It is easier to directly import the correlation subroutine using an alias (e.g. "do_correlation"):

```
from multipletau_cor_tttr.correlate import CCF as do_correlation
```

You can then use the imported function `do_correlation` directly as:

```
cor,stdcor,timeaxis = do_correlation(data1,data2)
```

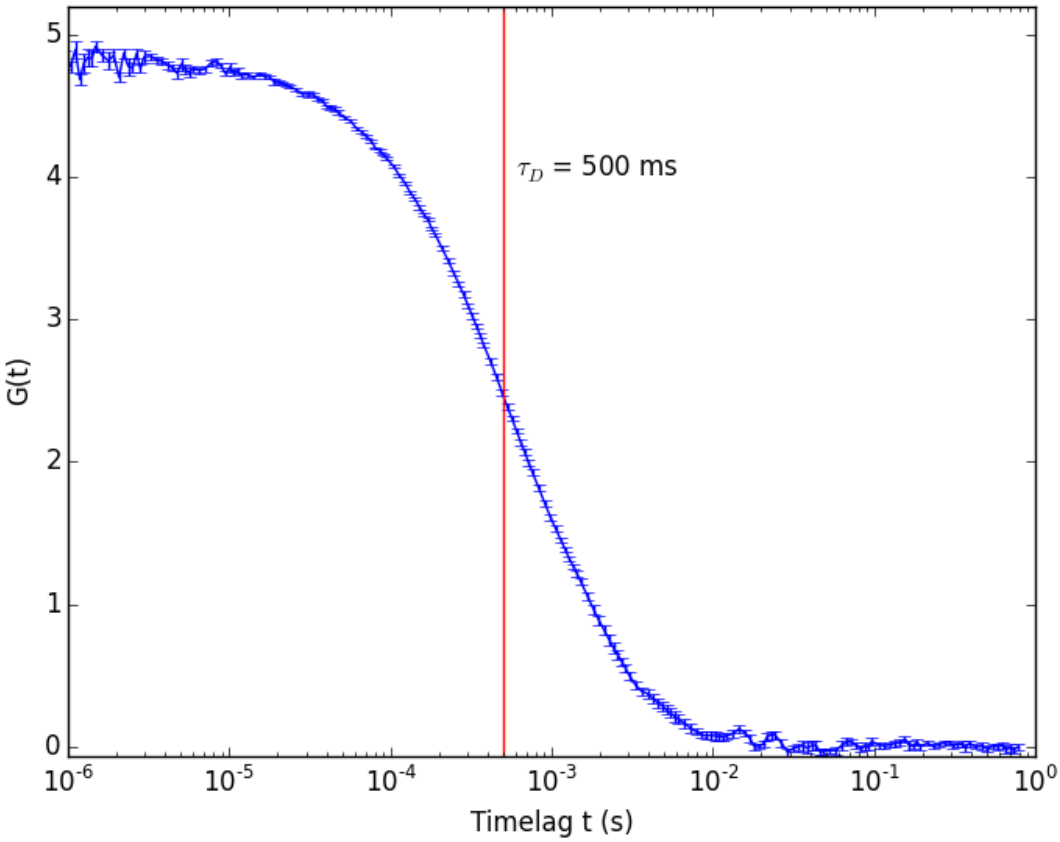
Here, `data1` and `data2` are the photon time stamps in channel 1 and 2, and the resulting correlation function is given by `timeaxis` and `cor`. Additionally, the standard error of mean (SEM) for every data point is given in `stdcor`.

1.3 Running the example script

Example data is provided with the program. It is found in the python /bin folder. Open a terminal and execute:

```
$ python multipletau_cor_tttr_example.py
```

Sample data is loaded from the associated file and correlated. The resulting correlation function is plotted.


τ_D = 500 ms
G(t)
Timelag t (s)
0
1
2
3
4
5
10^{-6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}
10^{-1}
10^{0}

CHAPTER 2

multipletau_cor_tttr.correlate Reference

multipletau_cor_tttr.correlate.**CCF**(*t1*, *t2*, *nblock=10*, *nc=10*, *nb='auto'*)

Performs crosscorrelation of time-tagged photon data t1 and t2 using semi-logarithmic timeaxis with nb logarithmic levels and nc equally spaced timebins per level. Error estimation is performed by splitting the measurement into nblock time segments of equal length and taking the standard error of mean. The returned array yields the correlation of intensity fluctuations, decaying to zero.

Parameters:

- t1: Numpy arrays of photon arrival times in channel 1 (integer type)
- t2: Numpy arrays of photon arrival times in channel 2 (integer type)
- nblock: Number of blocks used for error estimation. (Default: 10)
- nc: Number of time points per logarithmic level. (Default: 10)
- nb: Number of logarithmic levels. 'auto' takes the maximum possible lagtime to calculate nb.

Return:

- mcorr: 1d array of correlation result
- stdcorr: Standard error of mean of correlation result
- timeaxis: Timeaxis

multipletau_cor_tttr.correlate.**_CCF_inC**(*t1*, *t2*, *nc*, *nb*, *timeaxis*)

Wrapper function to communicate between python and C using ctypes library. The returned array yields the correlation of intensity fluctuations, decaying to zero.

Parameters:

- t1: Numpy arrays of photon arrival times in channel 1 (integer type)
- t2: Numpy arrays of photon arrival times in channel 2 (integer type)
- nc: Number of time points per logarithmic step
- nb: Number of logarithmic steps
- timeaxis: Logarithmic timeaxis as defined by nc and nb

Return:

- corr_res: 1d array of correlation result

CHAPTER 3

Indices and tables

- genindex
- modindex
- search

m

Symbols

C

M