

OpenTIMS TimsPy TimsR: Open Access to timsTOF Pro Raw Data

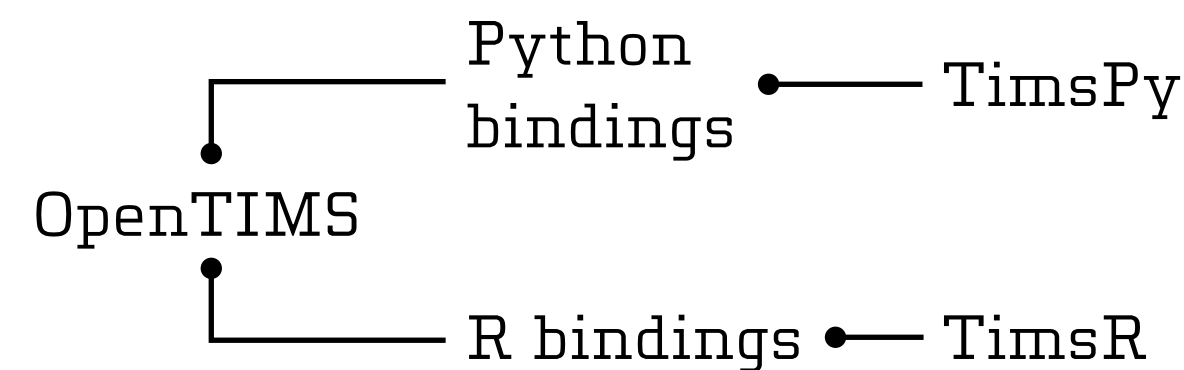


Mateusz K. Łacki¹, Michał P. Startek², Sven Brehmer³, Christina Bell³, Ute Distler¹, Stefan Tenzer¹

¹University Medical Center of the Johannes Gutenberg University Mainz, ²Institute of Informatics, University of Warsaw, ³Bruker Daltonik GmbH



OpenTIMS, TimsPy, and TimsR represent a modular and lean access to the RAW data. Minimal functionality is assured by using cushion bindings with small dependency basis. Higher order libraries offer additional functionality aiming to flatten the learning curve and provide basic quality plots. All tools are **open source**.



- The new Bruker's timsTOF Pro pairs Trapped Ion Mobility Spectrometry (TIMS) with liquid chromatography (LC) and mass spectrometry (MS).
- The instrument performs ion intensity measurements over time.
- Time is measured at 3 different scales that correspond to 3 different properties of ions:
 - base level: the time of flight of an ion corresponds to its mass to charge ratio
 - middle level: one push of ions from the second TIMS device is called a scan. Scan values correlate well with Collision Cross Section of ions and reflect their mobility.
 - top level: one full cycle of scans is called a frame. Several frames are needed to cover one chromatographic peak. Frame numbers are thus a natural unit to measure LC retention time.
- The three units of time offer a natural sparse representation of the data.

- **OpenTIMS** is a C++ library that provides access to timsTOF raw data format.
 - it is entirely open source
 - it does not use third party binaries
 - it is fast and can be optimised for specific computer architecture
 - direct bindings to OpenTIMS are provided for R and Python languages. To get those:

R: `install.packages('timsr')`

Python: `pip install timsPy`

- **TimsPy** and **TimsR** provide a simple access to Bruker's raw data.

```
from timsPy.df import TimsPyDF
D = TimsPyDF('path/to/data_folder.d')
```

```
library(timsr)
D = TimsR('path/to/data_folder.d')
```

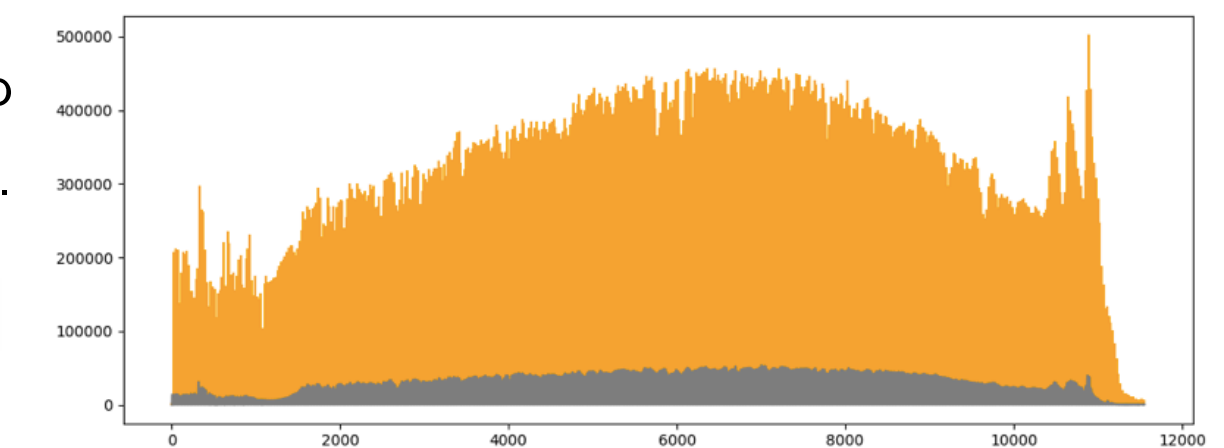
- Data is returned in a tabular form:

```
In: D[2] # selecting second frame
Out:
  frame scan  tof  intensity
0      2   33  97298         9
1      2   33 310524         9
2      2   34 127985         9
...    ...  ...   ...         ...
6595   2   912 338152         9
6596   2   913  56442         9
6597   2   915 172202         9
[6598 rows x 4 columns]
```

```
> D[2]
  frame scan  tof  intensity
0      2   33  97298         9
1      2   33 310524         9
2      2   34 127985         9
...    ...  ...   ...         ...
6595   2   912 338152         9
6596   2   913  56442         9
6597   2   915 172202         9
```

TimsR and TimsPy also offer basic quality plots.

```
D.plot_peak_counts()
```



Check out our github website for more information:

www.github.com/michalsta/opentims

www.github.com/MatteoLacki/timspy

www.github.com/MatteoLacki/timsr

Next special issue of the Journal of Proteome Research will contain an article where we describe the software in detail.