Manual for AlphaTims' command-line interface

Developed by: *Eugenia Voytik, Sander Willems and Matthias Mann*.

This manual is intended to illustrate the usage of the command-line interface (CLI) of AlphaTims.

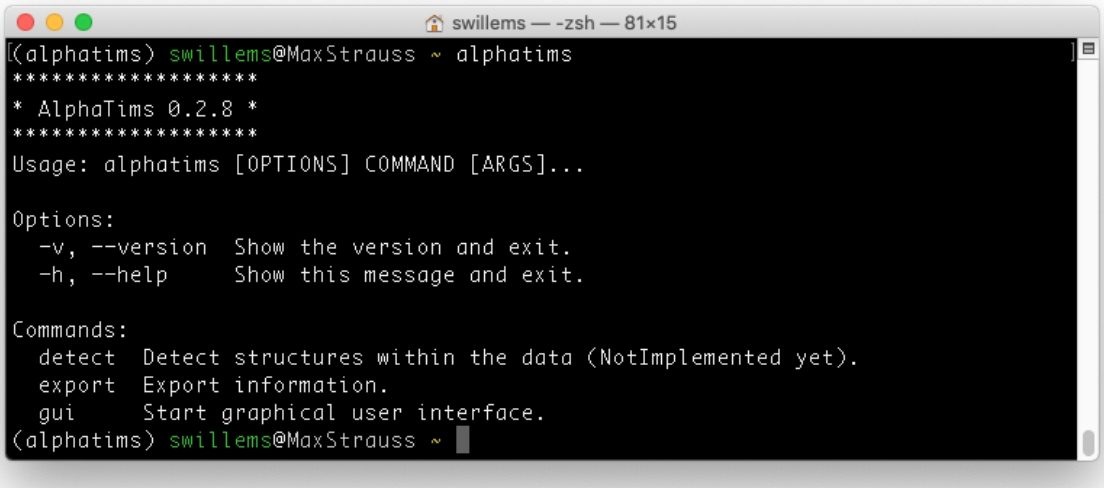
Table of contents

About

AlphaTims is an open-source Python package that provides fast accession and visualization of unprocessed LC-TIMS-Q-TOF data from Bruker's timsTOF Pro instruments. It indexes the data such that it can easily be sliced along all five dimensions: LC, TIMS, QUADRUPOLE, TOF and DETECTOR. It was developed by the Mann Labs at the Max Planck Institute of Biochemistry and is available as a freely available open-source tool with an Apache License and third-party licenses.

How to use the AlphaTims CLI

When AlphaTims is installed in a conda environment (see on installation section on GitHub), it always needs to be activated first with `conda activate alphatims`. Once this environment is active (indicated in the command-line by an `(alphatims)` prefix), the command-line interface (CLI) of AlphaTims can be run in a terminal with the command `alphatims`. This will open up the following window:



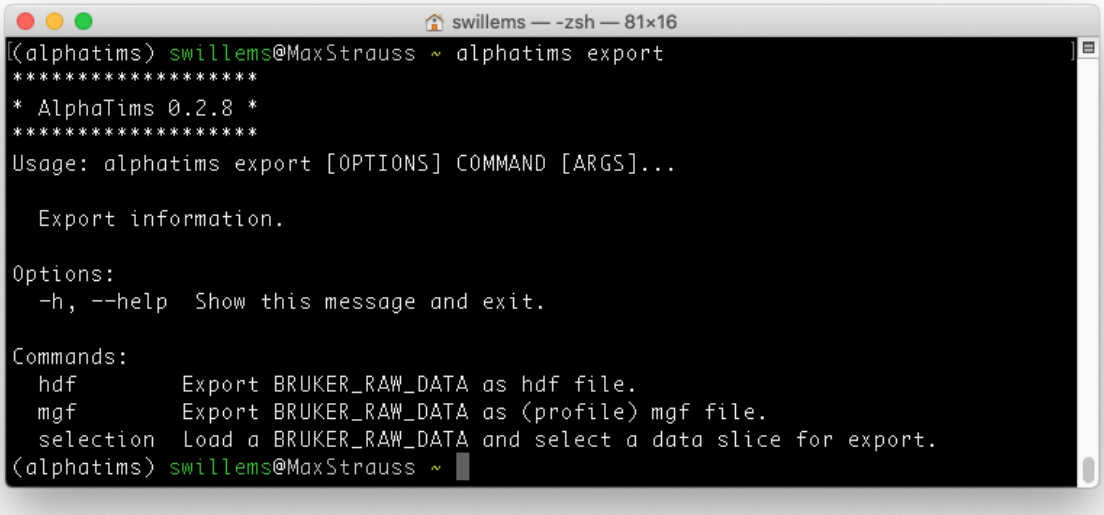
This main view shows that AlphaTims is correctly installed and able to run, as well as its version.

TROUBLESHOOTING:

- At any point, you can use the `-h` flag to get help about any command. Alternatively, you can try to find an answer in this manual or on the troubleshooting section of GitHub. If this still does not help, you can open an issue on GitHub.

AlphaTims provides three different options:

- The command `alphatims gui` will start the graphical user interface (GUI), which has its own manual.
- The command `alphatims detect` currently is unavailable, but will be implemented in the future.
- The third command `alphatims export` allows to export data and is the main focus of this manual. When run, the following output is shown:

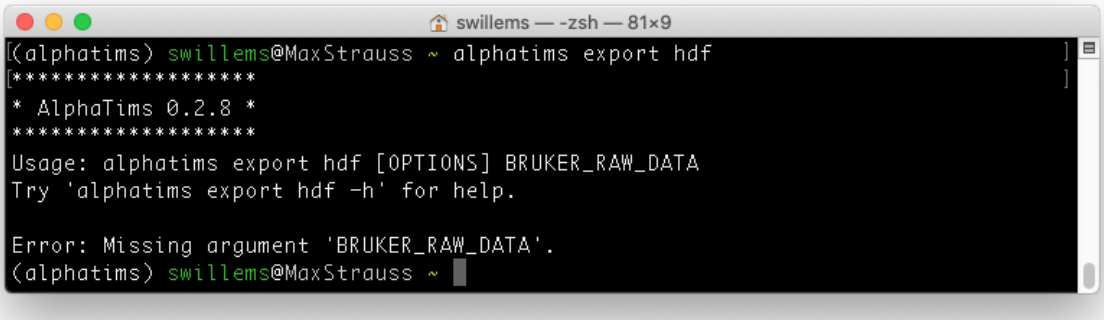


This provides three different options

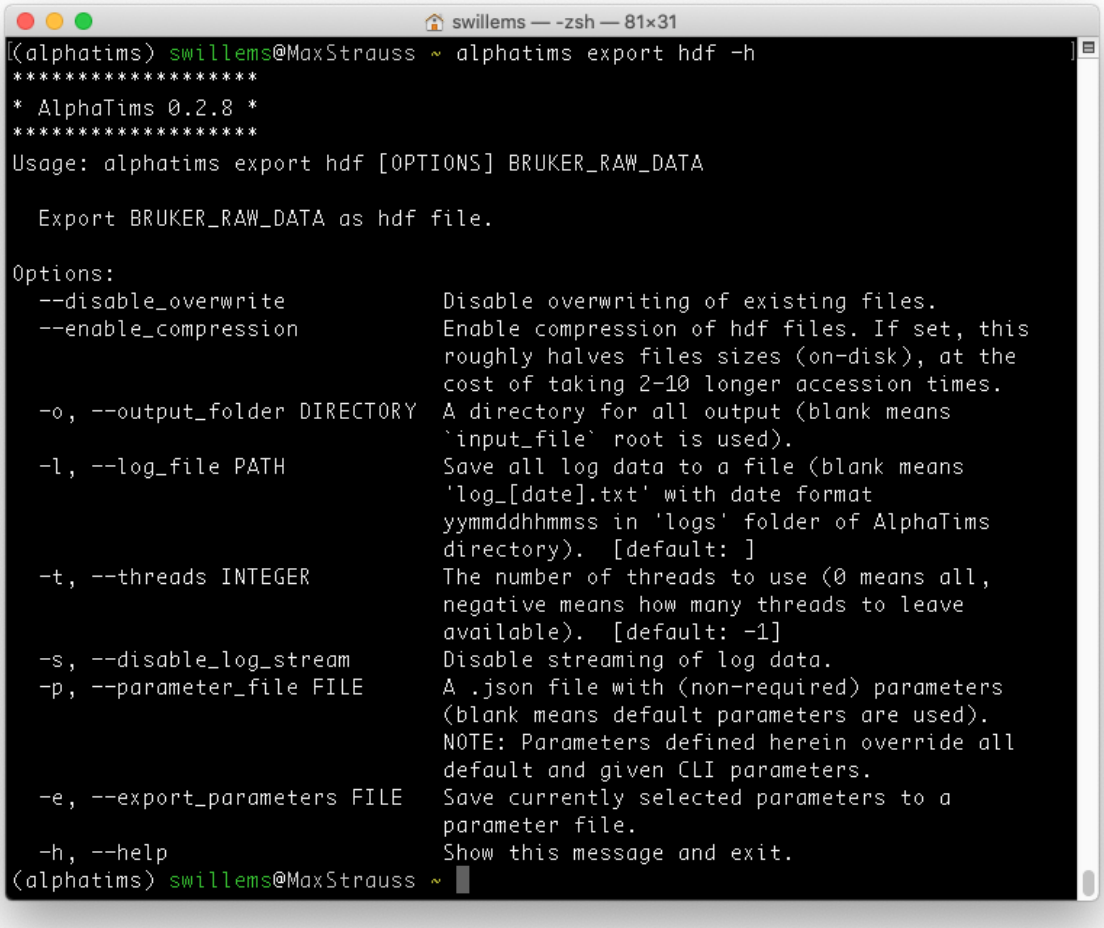
- alphatims export hdf, to index the raw data and save it as a portable HDF5 file.
- alphatims export mgf to create a traditional MGF file with MS2 spectra.
- alphatims export selection to export a selection of the data in tabular or graphical format.

Exporting an HDF5 file

The command alphatims export hdf has no further subcommands. It does however always require BRUKER_RAW_DATA input. As such, running this bare command provides the following error message:



Instead of running the bare command that is missing required input, you can use a -h flag to check out the full options:

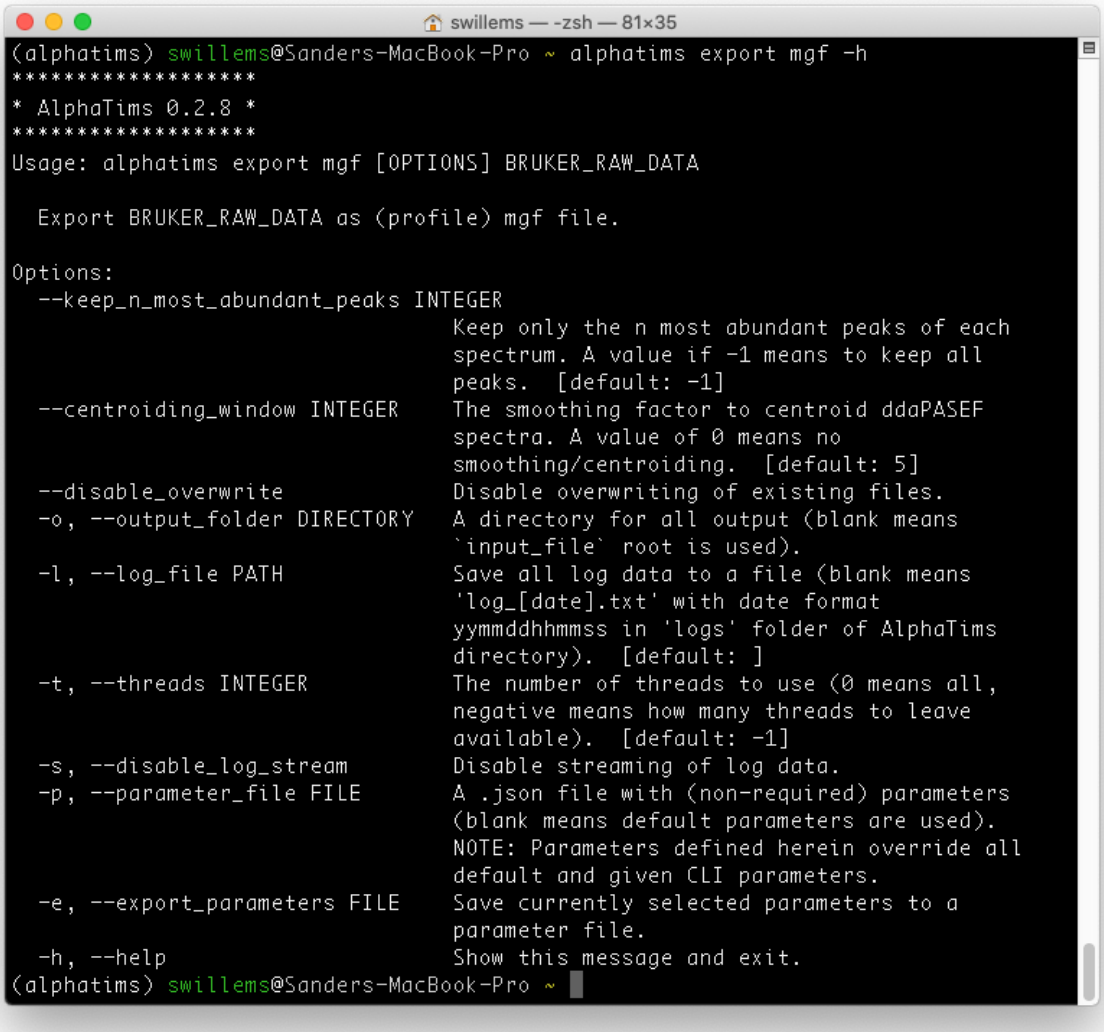


The first line states how to use this command correctly: alphatims export hdf [OPTIONS] BRUKER_RAW_DATA. There are a few important options that can be provided:

- --disable_overwrite: By default, AlphaTims converts the RAW_BRUKER_DATA file to a file with the same name, but with an ".hdf" extension instead. This overwrites any file if it already exists. By setting this flag, you can ensure you do not accidentally overwrite files that you previously created.
- --enable_compression: Saving raw data as an HDF5 file take roughly 6 Gb of disk space per billion detector events. By setting this flag, this file is automatically compressed which roughly halves the file size. Do note that saving and loading of decompressed files can be quite slow. Also note the when the provided BRUKER_RAW_DATA is already an HDF5 file, it can be (de)compressed, and that this option thus primarily exists to simplify file transfer and archiving.
- All other options are generic and are described in the "General options" section below.

Exporting an MGF file

This command is very similar to the alphatims export hdf command and also requires BRUKER_RAW_DATA input. Running it with a -h flag like alphatims export mgf gives the following output:



Similar to exporting HDF5 files, there are a few options available:

- --keep_n_most_abundant_peaks: TOF spectra can be quite noisy and contain many peaks. With this command you can ensure that only the most relevant peaks are retained.
- —centroiding_window: Minor fluctuations in TOF mean that an ion is detected in a small *m/z* range around its actual *m/z* value. This parameter determines how wide this range

can be. Smaller values will result in more fine-grained peaks, which potentially splits the signal from a single ion into two peaks. However, larger values lower the resolution.

- --disable_overwrite: By default, AlphaTims converts the RAW_BRUKER_DATA file to a file with the same name, but with an ".mgf" extension instead. This overwrites any file if it already exists. By setting this flag, you can ensure you do not accidentally overwrite files that you previously created.
- All other options are generic and are described in the "General options" section below.

Exporting a selection

The CLI also allows to export data selections with the command alphatims export selection. As with the other export commands, this always requires BRUKER_RAW_DATA input. Instead of running the bare command, it is best to use a -h flag to check out the full options.

There are several options to export a data selection:

1. **Coordinate selection**

   Coordinates in all dimensions can be selected. For each dimension, a (start, stop) pair is always required. If not provided for a particular dimension, the whole range is always taken.

   - --ion_type: This parameter accepts "precursors" or "fragments" as options. If both are required, the flag should be repeated, i.e. --ion_type fragments --ion_type precursors. The default is set to "fragments".

   - --rt_bounds: The retention time coordinates (in minutes). If a pair of negative integers is provided, these are assumed to be frame indices instead of retention time values.

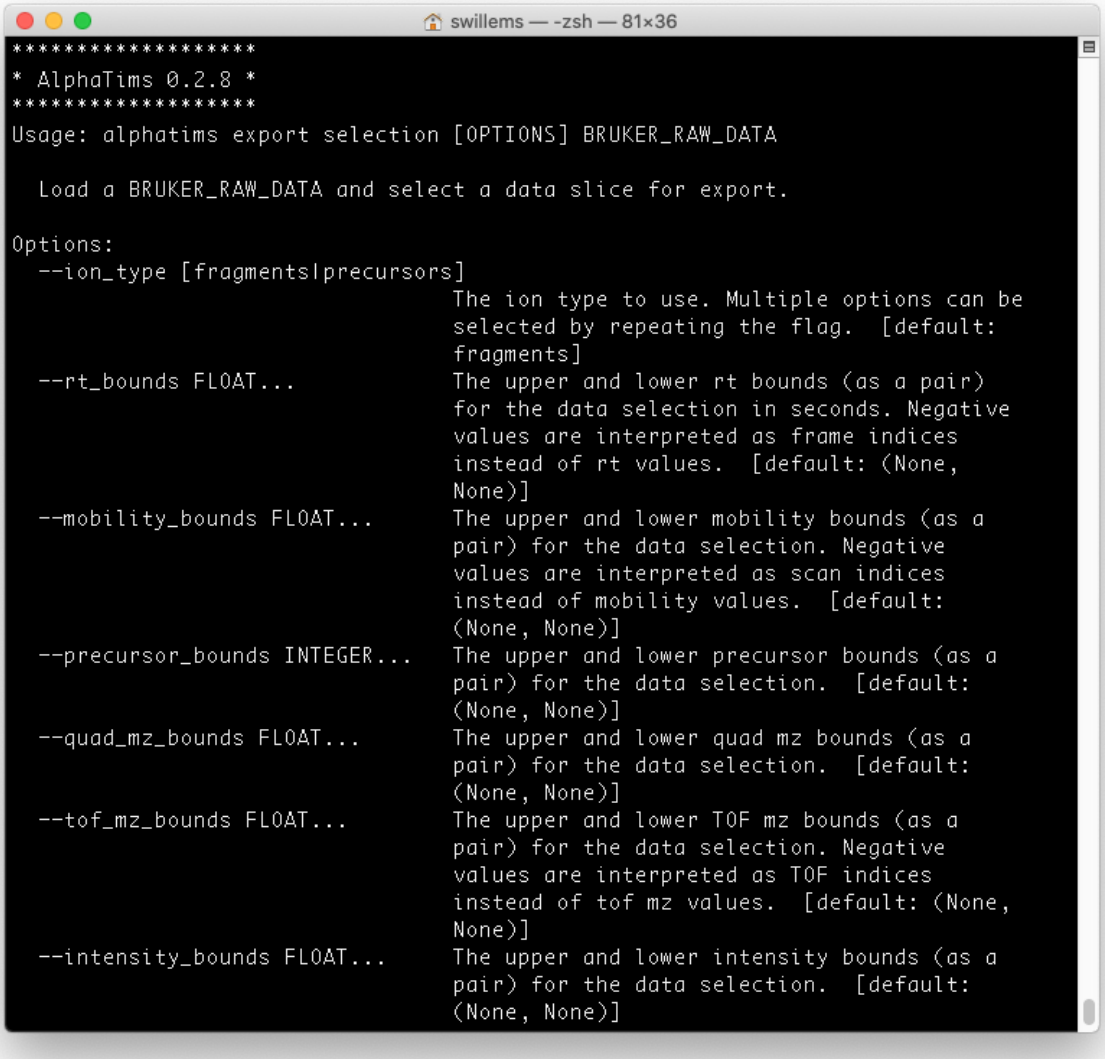
   - --mobility_bounds: The mobility coordinates (in $1/K_0$). If a pair of negative integers is provided, these are assumed to be scan indices instead of mobility values.

   - --precursor_bounds: The selected precursors. This is typically used to select individual spectra or DIA window groups. A single value can be provided by giving a consecutive pair. E.g., to select precursor 10 use --precursor_bounds 10 11. This has no effect if only precursors are selected.

   - --quad_mz_bounds: The quadrupole borders. This has no effect if only precursors are selected.

   - --tof_mz_bounds: The TOF *m/z* coordinates. If a pair of negative integers is provided, these are assumed to be TOF *m/z* indices instead of TOF *m/z* values.

- --intensity_bounds: The intensity values to select.

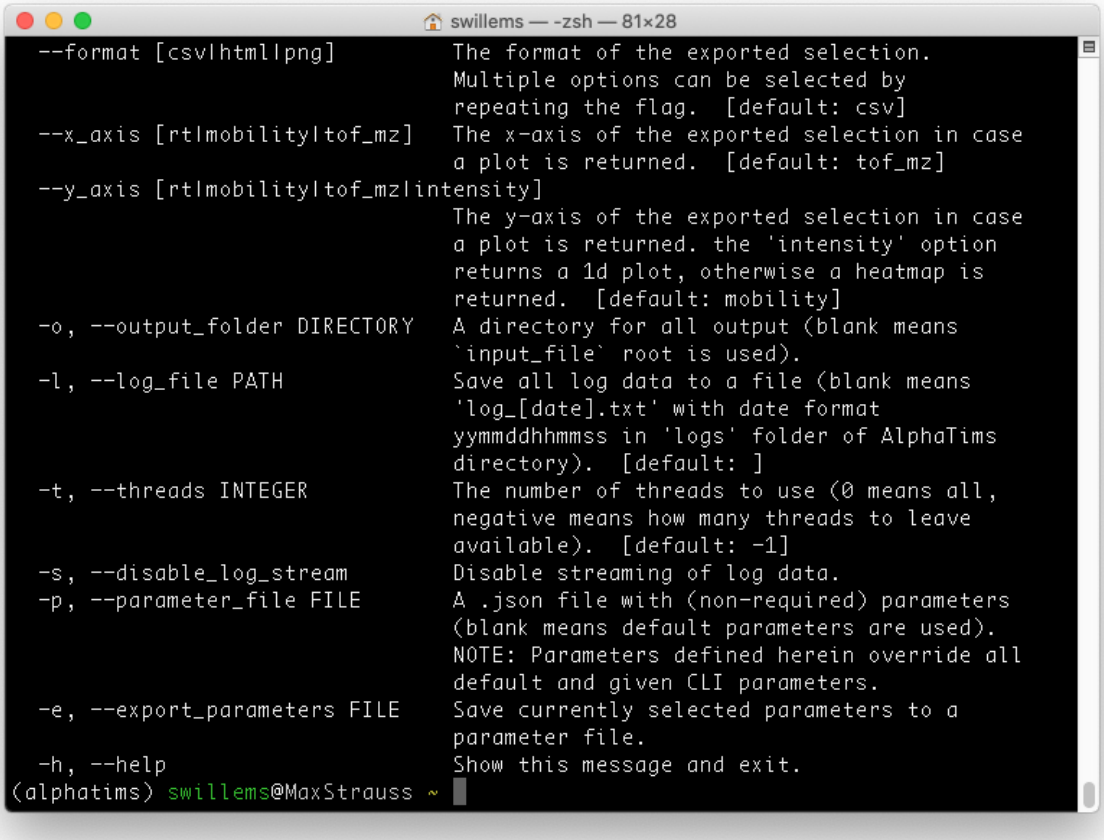


2. Export format

The CLI allows to export a data selection in three different formats with the --format parameter.

- CSV: A table with all coordinates. This is the default if no value is provided. A file is always exported with the same name is the input file, but with the extension "_selection.csv".
- HTML/PNG: A plot with the requested coordinates. A file is always exported with the

same name is the input file, but with the extension "_selection.html" or "_selection.png". If this option is selected, you can determine the axis with the --x_axis or --y_axis parameters.



General options

Most commands of the CLI come with some general options:

- -o, --output_folder: By default, AlphaTims uses the same folder as that of the BRUKER_RAW_DATA. This option allows to redirect the output to another folder.
- -l, --log_file: By default, AlphaTims saves all logs in its installation directory. This option allows to redirect the log file to another file and folder.
- -t, --threads: By default, AlphaTims uses all but one CPUs of your computer. This option allows to override that. Note that by providing a negative number you can instruct AlphaTims to leave these many threads unused.
- -s, --disable_log_stream: If many files are parsed with e.g., a script, the terminal output of AlphaTims can become tedious. You can suppress terminal output with this command.

- -p, --parameter_file: Instead of providing manual parameters, a file with (non-required) parameters can also be provided.
- -e, --export_parameters: To be able to reuse all used parameters for future commands, the current parameter selection can be exported with this command.
- -h, --help: Show a help menu instead of executing the command.