

Quickstart

Install SpiSOP and prepare the test datasets

Here is a **guide/tutorial** to get started with SpiSOP which was gratuitously provided by **Margo Willems** for download here: [SpiSOP Guide](#)

In short:

1. Download **SpiSOP basic standalone version (zip file)** (www.spisop.org/download)
2. Download **Matlab Compiler Runtime (MRC, 8.2 64bit for Windows)** (www.spisop.org/download)
3. Extract the **SpiSOP basic standalone version** to the **folder you want to install it to** (e.g. **D:\spisop_beta2.3**, recommended, since it then works out of the box)
4. Install the **Matlab Compiler Runtime (MRC)** on your computer and install it *irrespective* if you have Matlab already installed.
5. (*recommended*) Download 3 test datasets (www.spisop.org/download)
6. (*recommended*) Extract the test datasets to the main SpiSOP folder (e.g. **D:\spisop_beta2.3\test_datasets\...**)
7. (*optional*) run the browser for **sleep scoring** by running the *spisop_NOSETUP_just_open_file_for_sleep_scoring.bat* in the spisop folder, you can choose for example the **dummy dataset** in the subfolder *dummy* (e.g. **D:\spisop_beta2.3\dummy**).
8. (*optional*) For automated sleep scoring service in the browser from [Z3score](#) you can set the license permanently by changing the file *Z3Score_license.txt* (automated

sleep scoring is provided over the button **"import hyp"**

A few things to consider before actually starting with SpiSOP

If you use Windows make sure you can see the file endings in the explorer:

e.g. see:
<http://windows.microsoft.com/en-us/windows/show-hide-file-name-extensions#show-hide-file-name-extensions=windows-vista>

If you save files after you opened with a text editor, please make sure they are encoded in the format ANSI or alike (see https://en.wikipedia.org/wiki/Windows_code_page "UTF-8" does not work for the standalone version in every case) and the line endings are saved in PC/Windows/etc. mode and NOT Unix/Linux/Mac (FYI this is carriage return (CR) plus line fed (LF) or shortly written as `\r\n`).

Renaming of files (as described later) is important, especially for your convenience! This way you make sure that you touched all the necessary files and changed the parameters accordingly. This helps to exclude that you use the parameters of a previous analysis of a dataset. So, if you did not rename the files properly later those will create errors like "...file xxxx not found..." or "...could not open file xxxx..." ... which is good, so you do not perform an analysis under the false assumption of a previous dataset.

Better remember to check when you save or close a file if there is really no last or inbetween empty line(s), and if there is an empty line (or several ones) always remove them.

Check if correct line numbers of all datasets also corresponds

to their headers and hypnogram files etc.

If unsure about some function specific parameters (not the core parameters) read the description carefully and this documentation. Note that not all parameters are named fully matching this documentation. **If you are still unsure about the function specific parameters and also the default value does not help in understanding, then DO NOT DO THE ANALYSIS, because you do not know what you are doing here!**

Sampling rate for your datasets and **names of channels** can be usually found in header files of your data (for example for BrainVision files this is in *.vhdr files).

Do not forget to save the files you have opened (parameter files) after editing or closing.

Open the input files with a plain and simple text editor (NOT MS Excel or related), however the output files can be opened with a spread sheet program. For opening the output files it is recommended to use LibreOffice Calc <http://www.libreoffice.org/> (or OpenOffice Calc), or under circumstances also the "text import" function von MS Excel. For MS Excel however note in the German/Brasilian... Version that the decimal delimiter sign is actually a comma, thus this has to be considered for importing and saving any (input or) output data.

If you run SpiSOP, especially the browser, NEVER hit "Ctrl+C" key combination, as this terminates the program, and in the browser unsaved things are lost. (there is an autosaved version of the hypnogram if it was exported at least once, the default is every 5 changes the hypnogram is backed up). This cannot be changed, as matlab seems to have no (good) handle for this.

Setup SpiSOP for test analyses on test datasets and run it

SpiSOP will run a test analysis over command line by double clicking *spisop_test_run.bat* . You should NOT run spisop.exe without parameters. To set up the analysis you need to open the spisop_test_run.bat with a text editor (right click on this file, then choose "edit")

within you should adapt the line "SET pathPrefix=D:\spisop_toolbox_beta2.3" to the folder where you installed SpiSOP (i.e. where the spisop.exe is located in)

In the installation folder there is an "input" folder with subfolders for the three test analyses (e.g. ~/input/test_short ~/input/test_lindev ~/input/test_EMSA).

For all three subfolder you should adapt three files describing where the data (i.e. the test data you downloaded is located in). For the /input/test_short the three files are to be modified to have the correct file pathes to your data.

DataSetHeaderPaths_short.txt

DataSetPaths_short.txt

Hypnograms_short.txt

Do the same for equivalent files in ~/input/test_lindev and ~/input/test_EMSA

Don't forget to save the changes in the edited files!

If everything is set up correctly you can double click *spisop_test_run.bat* and in a command line window the test analyses will be run, one by one (this might take some time to run through)

The result files (spreadsheets) are written in the "output" folder within the installation folder (under respective

dataset names).

To alter the analyses look how the script is set up for those test datasets to run a specific analysis using the correct main functions.

A detailed tutorial and videos will follow soon, so please be patient, or contact spisop@spisop.org

...For running the SOURCE:

1. Open the file `~/spisop_toolbox/spisop_init.m` in Matlab and do the following:

- adapt the path prefix to the toolbox (consider unix or windows path notation) e.g.:

```
                                "pathPrefix =  
'/home/toolboxes/any/path/further/spisop_toolbox';"
```

- enable or disable the number of parallel "workers" by adapting the parameters:

```
                                parallelComputation = 'yes'; % either  
'yes' or 'no' default 'no'
```

```
                                numParallelWorkers = 4;% number of  
workers ...
```

- rename to your wished dataset name matching the input folder name you previously created

```
datasetAndInputFolderName = 'MyStudy';
```

2. Run the `spisop_init.m` line by line (or section by section) in Matlab (select lines, then hit F9 for Windows and Unix). The file is pre-set for example analysis of the test datasets. Alternatively one can run the "Sections" of the files.

For each new analysis be sure to start running lines in `spisop_init.m` always from beginning on!

The main functions of SpiSOP

There are 13 main functions in the SpiSOP toolbox:

1. ***hypvals*** – sleep values from sleep scorings (can be used for creating a sleep table) and export extract subsection of hypnograms with respect to sleep onset or start of hypnogram
2. ***freqpeaks*** – determine peaks in power spectrum bands of e. g. spindles or S0 etc. (depends on parameters)
3. ***spd*** – spindle or other fast oscillating events detection
4. ***sod*** – slow oscillation (S0) or delta wave or related events detection
5. ***pow*** – determine (average) power and power density (and energy) of specific frequency bands
6. ***nonEvents*** – find unique non-overlapping matching non-events (like non-spindles) of specified length and distance to corresponding real events of a corresponding sleep stage and in near proximity of those real events.
7. ***eventCooccurrence*** – discover co-occurrence of test and target events, i.e. if test (seed) events fall within a defined time window around target events.
8. ***confounds*** – reprocess hypnograms and exclude epochs with artifacts in a channel (e.g. EMG artifacts) or reduce epoch length (e.g. from 30 to 10-s epochs)
9. ***hypcomp*** – compare scoring of hypnograms statistically across scorers and create consensus scoring as well as export all hypnograms figures
10. ***browser*** – visualize the data in a data browser and do assisted sleep scoring and rechecking of data (also includes functionality of preprocout)

11. *preprocout* – preprocess the data and convert to other data formats (linear deviation, rereference, filtering)
12. *manipulateparam* – manipulate parameters of an parameter file, e.g. for repeated execution with different parameters
13. *remsMaAd* – detect rapid eye movements (REMs) during sleep (in particular REM sleep), Thanks to Marek Adamczyk (publication of him and others is in preparation)

How to run the main functions with SpiSOP in batch mode

The program is executed in the comandline (cmd.exe, bash, etc.) also using batch scripts.

For an example for windows see below the usage and the example run script *spisop_test_run.bat* in the main folder of the toolbox that will execute a testrun if SpiSOP and all three test datasets (i.e. test_short, test_EMSA, test_lindev) have been set up correctly, see question ***How to install and initialize datasets?***

USAGE (e.g. windows):

```
spisop.exe functionName currentFullInstallationFilePath  
inputFolderName outputFolderName coreParamFile  
functionParamFile
```

REQUIRED PARAMETERS:

functionName – is the name of the function to call e.g. hypvals spd sod ...see below

currentFullInstallationFilePath – is the path to the toolbox directory e.g.

D:\spisop_toolbox_beta2.3

inputFolderName – is a name to the subfolder within the input folder where the parameter files are stored

outputFolderName – is a name of the folder (to be created) in the general “output” folder to place the output files

coreParamFile – is the full name of the file containing the core parameters stored in the input folder within a subfolder given by *inputFolderName*

functionParamFile – is the full name of the file containing the function parameters stored in the input folder with *inputFolderName*

FUNCTIONS for ***functionName*** are:

hypvals

preprocout

browser

sod

pow

freqpeaks

confounds

nonEvents

hypcomp

eventCooccurrence

manipulateparam

remsMaAd

SPECIAL CASE for manipulateparam: here the parameters are:

```
spisop.exe manipulateparam currentFullInstallationFilePath  
inputFolderName      outputFolderName      inParamFilename  
outParamFilename param value_str
```

..., where

outputFolderName will be ingnored,

inParamFilename is paramter file name and outParamFilename the parameter file to write out to,

param is the parameter name to manipulate,

value_str gives the value to overwrite the old paramter value

OPTIONAL PARAMETERS as “[parameter]=[value]” pairs:

prefixExtentionName=[value] give a prefix extention string attached to outputfiles to the standard prefix

parallelComputation=[integer >0 and < 13] give number of workers for parallel processing to initiate

NOTE: For examples of coreParamFile and functionParamFile see the respective files for the test datasets of the is given in the “input” subfolders of the toolbox and referenced by the spisop_test_run.bat. See below on how to change the parameters in those files and what is necessary to set up other datasets and analyses.

...In SOURCE:

those functions are preceded by “spisop_” (e.g. spisop_hypvals(...))

Each function (with exception to spisop_eventCooccurence) has

5 input parameters:

1. path to the input folder with further input core parameter and parameter files
2. path to the output folder where to store the output
3. a prefix for all output files added in front of it
4. the text file with the core parameters for datasets, headers, hypnograms and filter settings
5. the text file with the parameters for the function

e.g. `"spisop_spd(pathInputFolder, pathOutputFolder, 'test_', 'CoreParam.txt', 'SpindlesParam.txt');"`

Since the `spisop_eventCooccurrence(...)` does not rely on core parameters, it only has four parameters:

e.g. `"spisop_eventCooccurrence(pathInputFolder, pathOutputFolder, 'test_', 'EventsCooccurrenceParam.txt');"`

What else to prepare for other datasets?

An example for test datasets of the following files is given in the "input" subfolders of the toolbox

1. Again, you need the path to the SpiSOP toolbox folder (e. g. "D:\my_toolbox\spisop_toolbox") (please avoid the use of spaces in the filepath)
2. For each subset of channels you want to analyze a file listing those channel names for each dataset
3. Absolute path in the filesystem to each of the files (e.g. eeg file, header file)
4. File containing the sample time-points of "Lights Off" marker
5. File with a list of frequency bands (example is given

for the test datasets).

6. File containing “center frequencies” for spindle (or slow oscillation) detection (can be gained from the `spisop_freqpeaks` function of the toolbox.

Hint: All filenames and filepaths should NOT contain commas (i.e. “,”) and avoid spaces (i.e. “ ”).

For further details see below or [Documentation](#).

INPUT of the main functions (Parameter files)

For help on the supported data format see Question “**What is needed and in what format?**”

In the input folder (e.g. `~/spisop_toolbox/input/test_EMSA/` for the test datasets) one can edit the text file with the parameters (e.g. `SpindlesParam.txt`), descriptions of the parameters are given in the file.

Parameters are loosely structured by importance for modification in

“REQUIRED” (please always modify or check),

“OPTIONAL” (consider the use and be aware of what it changes)
and

“ADVANCED” (you should definitely know what you are doing here!)

Note: All input text files better not contain an empty lines, also at the end of input text files empty lines should be

avoided, since all empty lines will be ignored. Empty lines could lead to an unintentional mix-up of which hypnogram belong to which dataset!

GENERAL REMARKS:

Never use quotation marks in the files nor use any further commas (use points instead).

You can *additional* uncommented lines by using two hashes and two commas e.g.:

```
## This is a comment line,,
```

Only change the parameter itself (not the parameter name or description), i.e., only change the value after the first comma given the possible parameters stated in the description)

In the parameter file you also give the names of the other files in the “input” folder with lists (e.g. list of dataset paths, header path, file with channels etc.)

CHANNELS:

If you use channel names you can mention them as described in the fieldtrip documentation of the *ft_channelselection* function:

http://fieldtrip.fcdonders.nl/reference/ft_channelselection

For example to consider all channels one can write *all* instead listing all channels like *Cz,Fz,F3,...*

or use *C*,F** for all channels starting with C and F (e.g. *Cz,C4,F3,F4...*). Note that the Channels are given case sensitive, i.e. if *f** does not match the channel *F3*, however *F** will match *F3*.

ENTERING VALUES:

For giving numbers one can use one of the following notations:

e.g. 3 3.0 0.3456 -20.45 -3E-6 -3.99E6 4.6E2 -Inf Inf 0 are all valid, -1.45E-5 means -1.45 multiplied by 10^{-5}

OUTPUT of the main functions (Result files)

The output is stored in files with comma separated values ("*.csv", http://en.wikipedia.org/wiki/Comma-separated_values) and ASCII compatible characters (http://de.wikipedia.org/wiki/American_Standard_Code_for_Information_Interchange) in the folder output folder (e.g. ~/spisop_toolbox/output/test/)

Be aware that some of the values are in decadic exponential notation as e.g. 3.55e-10 and need to be recognized as such when open by spread sheet processors like MS Excel or OpenOffice/LibreOffice Calc (***select the option "Detect special numbers" for text import, see https://help.libreoffice.org/Common/Text_Import#Detect_special_numbers***).

Often when there are problems when reading and opening csv files in spreadsheet software of some countries (e.g. Germany, Brasil), especially with Microsoft® Excel®. Since in those countries the comma is reserved for a separator of decimal values (e.g. 2,56 instead of 2.56) the common practice is that the values are separated by a semicolon instead of a comma. For Germans readers one can refer to <http://blogs.system-worx.de/kejr/2010/12/04/csv-dateien-und-excel-ein-deutsches-problem-mit-losung/> for an explanation and how to deal with that. Results can be easily converted to native *.xls files using converter programs (e.g. "Convert XLS"

<http://www.softinterface.com/Convert-XLS/Convert-XLS.htm> or
“Visual Text Converter”
<http://www.heise.de/download/download/visual-text-converter-1176687.html>
l) or just opening in LibreOffice and “save as” *.xls file.

Be aware that if data will be down sampled, then sample values in the output data refer to the down sampled data points, NOT the original samples, therefore also the sample rate invariant time in seconds of each point is given in extra columns.

The word “trough” refers to a negative peak in potential of the filtered signal (,i.e. pre-filtered for analysis and not the raw signal, e.g. in slow oscillations with a band pass filter of 0.3 to 3.5 Hz), the word “peak” to the equivalent positive peak. Therefore the down-peak of the for example a slow oscillation event is the “maximal trough” and the up peak afterwards is the “maximal peak”. For spindles the “maximal peak” does not need to be after the “maximal trough”.

What should I check before running or editing the parameters of the functions?

- The unit and of your recorded **potential** in the data e.g. μV , V, fT, T etc. (often EEG is given in μV therefore all values in the input and output need to refer to μV where the word “potential” is mentioned, μV is assumed as the default)
- Header files and if they are accessible, look at the information within them if possible (e.g. in Brainvision header files one get information about previous filtering, channels, resolutuoin, samplingrate etc.,

electrode impedances)

- Sample frequency of your data ...and if they match or get the highest common frequency between datasets (sample frequency is often found in the header files)
- Sample frequency is at least 3 times (better is four times) the maximal frequency of interest (e.g. spindles 16 Hz therefore at least $3 \times 16 \text{ Hz} = 48 \text{ Hz}$ but better more than 64 Hz sample rate is needed, however a minimal 100 Hz is recommended which gives later a 10 ms precision). That is that the Nyquist-Shannon sampling theorem holds: ([http://en.wikipedia.org/wiki/Nyquist-Shannon_sampling_theorem](http://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem)), i.e. the sampling frequency of the data f_s must be higher than the maximal frequency of interest. For example if your sampling frequency is 200 Hz then the maximal frequency you can include in your analysis is therefore $< 100 \text{ Hz}$, i.e., 99 Hz. Note however that in practice the sampling frequency should be more than two-fold (e.g. 2.2) and here more at least three-fold the maximal frequency of interest since there is no perfect low pass filter.
- List of lights off marker samples → open manually the header files to check for correctness of markers (this is not automatically extracted from marker files on purpose), if not accessible use 0 for those files to signify that the lights off marker begins with the first sample of the recording. Lights off samples should be given as the sample in corresponding to the original read in dataset sampling frequency (,i.e., before it is downsampled for analysis)
- Previous applied filters (e.g. during recording) of the data, like high cut-off or low cut-off filters. This can be looked up by opening the header files in a text editor. Note in brain vision file headers the low cut-off filters can be obtained by the low cut-off frequency $f_{\text{lowcut}} = 1/(2 \times \pi \times \tau)$, where τ is the time constant in seconds and π approx. 3.14159, e.g. $\tau = 1 \text{ s}$ therefore $f_{\text{lowcut}} = 0.159 \text{ Hz}$

- What is a negative and what is a positive signal, especially if you use edf files, sometimes even if the values in the file are stored as negative (e.g. -16.3) this can mean that the actually measured potential is positive (e.g. 16.3). So be aware of that, this depends on your recording system or (export) software used. SpiSOP assumes that the values in the file are correct, however one can multiply each dataset (or all datasets) with a specific factor (e.g. -1) for inversion for such known cases (see core parameters). To decide in case you have no clue, what you can do is view some sleep-EEGs in a viewer-software (better not one of the manufacturer, e.g. <http://www.teuniz.net/edfbrowser/>) and see if K-complexes and S0s show characteristic steep negative Potentials before going into positive (if it goes consistently first up then down this is an indicator that positive and negative are mixed up!).

Tips and Hints:

Always store the parameter file and the files given in it with the results, so it is clear how the analysis was done and how to later report the results, e.g. for publication.

When scoring the hypnogram try to mark also small movement arousals (MA) or stages with even small artifacts (e.g. >50 μ V in EMG) as MA to exclude later erroneous detection (e.g. for spindle detection).

When Lights-off is after the first sleep stage of interest, try to mark the previous epochs manually in the hypnogram as MA i.e. a "1" in the second column of the file.

When detecting spindles of S0 (or equivalent) first detect for individual channels (e.g. parameter *ThresholdAggregationMethod* should then be set to *respectivechan*) to find erroneous datasets or channels or

events by looking at (mean) trough to peak potentials or the (mean) standard deviation of the filtered signal. Bad datasets, channels or events have markedly different standard deviation or trough to peak potentials than other datasets, channels or events.