

## Quick guide to all the files of SPIKY

Note: For the Matlab m-files these short descriptions can also be obtained in Matlab by typing 'help <m-filename>.

### **Main Matlab-files**

These are files that might be used by the user directly.

**SPIKY\_compile\_MEX.m:** This is the first file that should be run once the zip-package has been extracted. You can find it in the folder 'SPIKY\_MEX'. Once the MEX-files have been compiled you can run the main program SPIKY.

**SPIKY.m:** Main function that starts the GUI. It opens one figure on the left that contains all the elements of the GUI.

**SPIKY\_f\_user\_interface.m:** This function consists of two parts. The upper part can be used to define some of the standard parameters of SPIKY. The lower part can be used to generate predefined spike train datasets which can then be called via the listbox in the 'Selection: Data' panel. Always make sure that the variable 'listbox\_str' labels correctly the datasets defined in the subsequent indices.

**SPIKY\_f\_pico.m:** This function can be used to obtain from piecewise constant measure profiles (that can be extracted via the red mouse button while hovering over a profile) the average value as well as the x- and y-vectors for plotting.

**Other Matlab-files (all these files can be found in the folder 'SPIKY\_m\_files')**

These are files that will be used within SPIKY and there should be no need for the user to use these file directly.

[SPIKY\\_calculate\\_distances.m](#): This calculates the selected time-resolved spike train distances once the button 'Calculate' is pressed. In case of very long datasets it also initiates the memory management.

[SPIKY\\_check\\_calculate.m](#): This doublechecks all inputs made in the SPIKY panel 'Selection: Measures'.

[SPIKY\\_check\\_plot.m](#): This doublechecks all inputs made in the SPIKY panel 'Selection: Plots'.

[SPIKY\\_check\\_spikes.m](#): This checks the input spikes, e.g., it eliminates spikes outside limits as well as double spikes. It also converts the input to the format used in SPIKY, e.g., cell arrays with spike times.

[SPIKY\\_check\\_update.m](#): This doublechecks all inputs made in the SPIKY panel 'Parameters: Data'.

[SPIKY\\_frame\\_comparison.m](#): This compares the dissimilarity matrices selected in the panel 'Selection: Measures' for different time instants and/or selected or triggered averages (once the 'Plot' button is pressed and if 'Frame comparison' is checked).

[SPIKY\\_frame\\_sequence.m](#): This initializes the sequence of frames / movie (once the 'Plot' button is pressed and if 'Frame sequence' is checked).

[SPIKY\\_get\\_select\\_ave.m](#): This is an example file that demonstrates how to set the intervals for the selective averaging using a Matlab script.

[SPIKY\\_get\\_instants.m](#): This is an example file that demonstrates how to set the time instants using a Matlab script.

[SPIKY\\_get\\_trig\\_ave.m](#): This is an example file that demonstrates how to set the triggered averaging using a Matlab script.

[SPIKY\\_handle\\_change\\_font\\_position.m](#): This function which is called by 'SPIKY\_handle\_font.m' allows changing the position (x and y) of text objects (such as axis or tick labels) in the figure.

[SPIKY\\_handle\\_change\\_property.m](#): This function which is called by several other functions allows changing properties of objects like lines, patches and texts in the figure.

**[SPIKY\\_handle\\_change\\_subplot\\_size.m:](#)** This function which is called by 'SPIKY\_handle\_subplot.m' allows changing the position (x and y) and the size (width and height) of the matrix subplots in the figure.

**[SPIKY\\_handle\\_edit\\_line\\_position.m:](#)** This function which is called by 'SPIKY\_handle\_line.m' allows changing the position of certain lines (such as markers and separators) in the figure.

**[SPIKY\\_handle\\_edit\\_string.m:](#)** This function which is called by 'SPIKY\_handle\_font.m' allows changing the string of text objects (such as axis labels or subplot titles) in the figure.

**[SPIKY\\_handle\\_edit\\_subplot\\_position.m:](#)** This function which is called by 'SPIKY\_handle\_subplot.m' allows changing the position of individual subplots in the figure.

**[SPIKY\\_handle\\_font.m:](#)** This function allows changing the string, the position, and various properties (such as color, font size, font weight, and font angle) of text objects in the figure.

**[SPIKY\\_handle\\_line.m:](#)** This function allows changing the properties (such as color, line style, and line width) of line objects in the figure.

**[SPIKY\\_handle\\_patch.m:](#)** This function allows eliminating the color coding for spike train groups (used for example in the dendrograms).

**[SPIKY\\_handle\\_set\\_property.m:](#)** This function which is called by several other functions allows changing properties (such as color, font weight, font angle, and font size) of matrix tick labels. It can be invoked by hovering above (not outside!) the very left part of the matrix subplot and pressing the right mouse button.

**[SPIKY\\_handle\\_subplot.m:](#)** This function which calls 'SPIKY\_handle\_change\_subplot\_size.m' allows changing the position (x and y) and the size (width and height) of the matrix subplots in the figure.

**[SPIKY\\_hints.m:](#)** This sets the short hints (tooltip strings) that are displayed when the mouse hovers above an element of the graphical user interface SPIKY (if 'Hints (Tooltips)' is selected in the 'Options' menu).

**[SPIKY\\_hints\\_no.m:](#)** This disables the short hints (tooltip strings) that are displayed when the mouse hovers above an element of the graphical user interface SPIKY (if 'Hints (Tooltips)' is not selected in the 'Options' menu).

**[SPIKY\\_hints\\_STG.m:](#)** Depending on whether 'Hints (Tooltips)' is selected in the 'Options' menu this does or does not set the short hints (tooltip strings) that are displayed when the mouse hovers above an element of the spike train generator (STG).

[\*\*SPIKY\\_matrices.m:\*\*](#) This initializes the matrices (in particular the selective and triggered averages) for the frame comparison and the frame sequence (once the 'Plot' button is pressed and if either 'Frame comparison' or 'Frame sequence (movie)' is checked).

[\*\*SPIKY\\_paras\\_get.m:\*\*](#) This sets the various parameter structures to the values of the various components of the graphical user interface SPIKY (somehow inverse to SPIKY\_init).

[\*\*SPIKY\\_paras\\_set.m:\*\*](#) This initializes the components of the graphical user interface SPIKY to the values stored in the various parameter structures (somehow inverse to SPIKY\_get\_paras).

[\*\*SPIKY\\_plot\\_allspikes.m:\*\*](#) This plots the spikes once they are loaded (either via 'Select from list', 'Spike train generator' or the 'Load' buttons).

[\*\*SPIKY\\_plot\\_frame.m:\*\*](#) This plots the matrices and dendrograms for one frame of the movie.

[\*\*SPIKY\\_plot\\_profiles.m:\*\*](#) This plots the time profiles of the selected measures (given that 'Dissimilarity profiles' is checked in 'Selection: Plots' and after the 'Plot' button has been pressed).

[\*\*SPIKY\\_plot\\_spikes.m:\*\*](#) This plots the spikes once all the measures have been calculated (after the 'Calculate' button has been pressed).

[\*\*SPIKY\\_plot\\_updatespikes.m:\*\*](#) This plots the spikes once the 'Parameters: Data' have been updated.

[\*\*SPIKY\\_select\\_instants\\_averages.m:\*\*](#) This function provides an input mask (for keyboard and mouse) for selecting time instants as well as selective and triggered averages (intervals and time instants, respectively) which will make up the individual frames of the movie.

[\*\*SPIKY\\_select\\_markers.m:\*\*](#) This function provides an input mask (for keyboard and mouse) for selecting thick and thin time markers (lines denoting the times of specific events).

[\*\*SPIKY\\_select\\_separators.m:\*\*](#) This function provides an input mask (for keyboard and mouse) for selecting thick and thin spike train separators (lines dividing the spike trains).

[\*\*SPIKY\\_select\\_spike\\_train\\_groups.m:\*\*](#) This function provides an input mask (for keyboard and mouse) for selecting spike train groups (their names and their sizes).

[\*\*SPIKY\\_select\\_spike\\_train\\_groups\\_calc.m:\*\*](#) This function provides an input mask for the selection and sorting of your input spike train groups *before* the calculation of the dissimilarity measures. You can select/eliminate spike train groups and change their order manually via the buttons at the top.

[\*\*SPIKY\\_select\\_spike\\_train\\_groups\\_plot.m\*\*](#): This function provides an input mask for the selection and sorting of your input spike train groups *after* the plotting of the dissimilarity measures. You can select/eliminate spike train groups and change their order manually via the buttons at the top. Furthermore, once you have plotted some dendrograms you can also sort the spike train groups according to the clustering obtained.

[\*\*SPIKY\\_select\\_spike\\_trains\\_calc.m\*\*](#): This function provides an input mask for the selection and sorting of your input spike trains *before* the calculation of the dissimilarity measures. You can select/eliminate spike trains and change their order manually via the buttons at the top but you can also sort the spike trains according to some predefined criteria (such as number of spikes and latency).

[\*\*SPIKY\\_select\\_spike\\_trains\\_plot.m\*\*](#): This function provides an input mask for the selection and sorting of your input spike trains *after* the plotting of the dissimilarity measures. You can select/eliminate spike trains and change their order manually via the buttons at the top but you can also sort the spike trains according to some predefined criteria (such as number of spikes and latency). Furthermore, once you have plotted some dendrograms you can also sort the spike trains according to the clustering obtained.

[\*\*SPIKY\\_select\\_spikes.m\*\*](#): For very large datasets this function provides a way to reduce the number of spikes of your datasets even before the plotting of the spikes (which can take very long for a huge number of spikes). You can select/eliminate spike trains and change their order manually via the buttons at the top but you can also sort the spike trains according to some predefined criteria (such as number of spikes). Finally, you can restrict the time window.

[\*\*SPIKY\\_select\\_variable.m\*\*](#): This function allows to select the variable / field of a Matlab-file that contains the quantity (spikes, markers, separators etc.) to be loaded into SPIKY.

[\*\*SPIKY\\_set\\_options.m\*\*](#): This function is called via the SPIKY main menu and allows to set a few options during runtime. In particular, you can activate hints (short explanations that will be shown when you hover with your mouse over a GUI-element). It will also enable you to save and load parameter settings (note that this functionality is still under development and thus currently disabled).

[\*\*SPIKY\\_STG\\_setup.m\*\*](#): This initializes the spike train generator (STG).

### **Spiky-loop (all these files can be found in the folder 'SPIKY\_loop')**

The program '**SPIKY\_loop**' (which replaces the old 'SPIKY\_no\_plot') is complementary to the graphical user interface 'SPIKY'. Both programs can be used to calculate time-resolved spike train distances (ISI

and SPIKE) between two (or more) spike trains. However, whereas SPIKY was mainly designed to facilitate the detailed analysis of one dataset, 'SPIKY\_loop' is meant to be used in order to compare the results for many different datasets (typically in some kind of loop).

**SPIKY\_loop.m:** This is the main program in which the variables are set and from where the main function 'SPIKY\_loop\_f\_distances' is called. You can plug in your dataset(s) and then use more or less the full functionality offered by SPIKY (such as dissimilarity profiles and pairwise dissimilarity matrices for individual time instants as well as selective and triggered averages both for spike trains and spike train groups).

**SPIKY\_loop\_examples.m:** This is an alternative main program in which some examples are shown that demonstrate how to get the most out of 'SPIKY\_loop'.

**SPIKY\_loop\_f\_distances.m:** This is the function (called by SPIKY\_loop) in which the various spike train dissimilarities are calculated (using the MEX-files described below).

**SPIKY\_loop\_local.m:** This is an example application that examines how the shift of a single spike affects the three time-resolved and parameter-free measures of spike train synchrony (SPIKE-Sync, ISI and SPIKE). For context please look at Fig. 4 in Kreuz et al. (2011) [[PDF](#)].

**SPIKY\_loop\_matrices.m:** This calculates the pairwise dissimilarity matrices for individual time instants as well as selective and triggered averages both for spike trains and spike train groups.

**SPIKY\_loop\_surro.m:** This is an alternative main program from where the main function 'SPIKY\_loop\_f\_distances' is called as well. However, instead of running over different datasets it allows to compare the results obtained for one dataset against results obtained for spike train surrogates generated from that dataset. This can help to estimate the statistical significance.

**SPIKY\_loop\_trigger.m:** This is similar to 'SPIKY\_loop.m' only that you do not look at different datasets, rather you look at different epochs within the same dataset. The typical setup would be a continuous recording during which different stimuli are presented in some random order. The task is to separate the responses to different stimuli and align them to their trigger onset. This could be very useful in order to investigate neuronal coding.

### **SPIKY-Functions (all these files can be found in the folder 'SPIKY\_m\_functions')**

**SPIKY\_f\_all\_sorted.m:** This function is called from SPIKY\_calculate\_distances\_MEX to sort the subplots selected in the 'Selection: Measures' panel. It also eliminates empty subplots.

**SPIKY\_f\_average\_pi.m:** This function calculates the average profile for piecewise linear profiles (such as SPIKE). This includes piecewise constant (such as ISI) which should first be transformed via the function 'SPIKY\_f\_pico.m' to a format suitable for plotting.

**SPIKY\_f\_compute\_gauss\_smooth:** This function computes the Peri-Stimulus Time Histogram, smoothed with a Gaussian filter of kernel width "window". Copyright: Jude Mitchell, <http://www.snl.salk.edu/~jude/sfn2008/index.html>.

**SPIKY\_f\_convert\_matrix.m:** If needed, this function converts the input spikes into the input format used by SPIKY, namely a cell with the spike times of the different spike trains as different cell arrays. It works both for padded matrices with spike times as rows as well as with matrices of zeros and ones with the ones indicating the spike times.

**SPIKY\_f\_event\_detector.m:** This function provides an input mask for detecting discrete events in continuous data. The resulting events are then used as spike input in SPIKY.

**SPIKY\_f\_get\_dt.m:** This function extracts the best estimate of the sampling interval from the spike dataset itself in case none is provided by the user.

**SPIKY\_f\_get\_events.m:** This function detects discrete events in continuous data.

**SPIKY\_f\_lab.m:** This function automatically sets the range and the spacing of the axis ticks.

**SPIKY\_f\_measure\_profiles.m:** This function plots the individual time profiles of the selected spike train distances and calculates their average.

**SPIKY\_f\_moving\_average.m:** This function calculates the moving average (including values from both past and future).

**SPIKY\_f\_moving\_average\_f.m:** This function calculates the anti-causal moving average (including future values only).

**SPIKY\_f\_moving\_average\_p.m:** This function calculates the causal moving average (including past values only).

**SPIKY\_f\_moving\_average\_para.m:** This function calculates the moving average (including values from both past and future) stretching as far as possible at the beginning. Many rows are calculated in parallel.

**SPIKY\_f\_moving\_average\_para\_f.m:** This function calculates the anti-causal moving average (including future values only) stretching as far as possible at the end. Many rows are calculated in parallel.

[SPIKY\\_f\\_moving\\_average\\_para\\_p.m](#): This function calculates the causal moving average (including past values only) stretching as far as possible at the beginning. Many rows are calculated in parallel.

[SPIKY\\_f\\_moving\\_average\\_weighted.m](#): This function calculates the moving average (including values from both past and future) with asymmetric inwards averaging at the edges.

[SPIKY\\_f\\_moving\\_average\\_weighted\\_f.m](#): This function calculates the anti-causal moving average (including future values only) with asymmetric inwards averaging at the edges.

[SPIKY\\_f\\_moving\\_average\\_weighted\\_p.m](#): This function calculates the causal moving average (including past values only) with asymmetric inwards averaging at the edges.

[SPIKY\\_f\\_poisson.m](#): This function creates a Poisson spike train with a given length, firing rate and refractory time.

[SPIKY\\_f\\_selective\\_averaging.m](#): This function calculates the selective averages for both piecewise constant (e.g. ISI) and piecewise linear (e.g. SPIKE) functions.

[SPIKY\\_f\\_spike\\_train\\_surrogates.m](#): This function generates various kinds of spike train surrogates (which differ in the selection of properties that are maintained). Currently the properties maintained are either the spike number, or (in addition) the interspike interval distribution.

[SPIKY\\_f\\_triggered\\_averaging.m](#): This function calculates the triggered averages for both piecewise constant (e.g. ISI) and piecewise linear (e.g. SPIKE) functions.

[SPIKY\\_f\\_unique\\_not\\_sorted.m](#): This function transforms a vector into a vector which contains all its unique elements but keeps these elements in the order of their first / last appearance (i.e. without sorting).

### **Matlab-Functions (all these files can be found in the folder 'SPIKY\_m\_functions')**

[linkage.m](#): This Matlab-function creates a hierarchical cluster tree.

[dendrogram.m](#): This Matlab-function generates a dendrogram plot of the hierarchical binary cluster tree generated by the LINKAGE function.

### **GUI-Figure**



[SPIKY.fig](#): This defines the outer appearance of the figure (created by means of the Matlab function 'guide') making up the graphical user interface SPIKY (on the left part of the screen).

### **Matlab-MEX-files (all these files can be found in the folder 'SPIKY\_MEX')**

[SPIKY\\_udists\\_MEX.c](#): This c-file calculates for each spike in every spike train the distance to the nearest neighbors in all other spike trains.

[SPIKY\\_ISI\\_MEX.c](#): This c-file calculates the (piecewise constant) ISI-distance.

[SPIKY\\_SPIKEpico\\_MEX.c](#): This c-file calculates the piecewise constant SPIKE-distance.

[SPIKY\\_realtimeSPIKEpico\\_MEX.c](#): This c-file calculates the piecewise constant real-time SPIKE-distance.

[SPIKY\\_forwardSPIKEpico\\_MEX.c](#): This c-file calculates the piecewise constant forward SPIKE-distance (which might be useful in the context of triggered averaging).

[SPIKY\\_SPIKE\\_MEX.c](#): This c-file calculates the sampled SPIKE-distance.

[SPIKY\\_realtimeSPIKE\\_MEX.c](#): This c-file calculates the sampled real-time SPIKE-distance.

[SPIKY\\_forwardSPIKE\\_MEX.c](#): This c-file calculates the sampled forward SPIKE-distance (which might be useful in the context of triggered averaging).

Once you have run the compilation file 'SPIKY\_compile\_MEX' there will also be a mexw32-file for each of these c-files.

### **SPIKY-Documentation (all these files can be found in the folder 'SPIKY-Documentation')**

[SPIKY-FAQ.doc](#): This file answers all the frequently asked questions regarding the graphical user interface SPIKY.

[SPIKY-Elements.doc](#): This file describes all the individual elements of the graphical user interface SPIKY.

[SPIKY-Files.doc](#): This file containing a short description of all files contained in the SPIKY-package.

[STG-Elements.doc](#): This file describes all the individual elements of the Spike Train Generator (STG).

[STG-Flowchart.pdf](#): This flowchart gives an overview of the general structure of the Spike Train Generator (STG).

[SPIKY-Readme.doc](#): This file contains a disclaimer and the BSD license.

[SPIKY-loop.doc](#): This file contains a short description of the program 'SPIKY\_loop' which is complementary to the graphical user interface 'SPIKY' and meant to be used in order to compare the results for many different datasets (e.g. in some kind of loop).

### **SPIKY-Testdata**

[SPIKY\\_testdata\\_01.mat](#): This is a mat-file containing some test data (our standard example as a matrix with 0 and 1). It can be loaded using the Load-mat function on either the menu or the taskbar.

[SPIKY\\_testdata\\_ca.mat](#): This is a mat-file containing some test data (our standard example as a cell array). It can be loaded using the Load-mat function on either the menu or the taskbar.

[SPIKY\\_testdata\\_cont.mat](#): This is a mat-file containing some continuous test data. It can be loaded using the Load-mat function on either the menu or the taskbar. The data are from the example used in Fig. 10 of our 2013 JNeurophysiol paper [\[PDF\]](#).

[SPIKY\\_testdata\\_zp.mat](#): This is a mat-file containing some test data (our standard example as a matrix with zero padding). It can be loaded using the Load-mat function on either the menu or the taskbar.

[SPIKY\\_testdata.txt](#): This is a text file containing some test data (5 spike trains covering the unit interval with 10 random spikes each). It can be loaded using the Load-txt function on either the menu or the taskbar.

### **SPIKY-Standard-parameters (all these files can be found in the folder 'SPIKY\_parameters) – [currently not used]**

[SPIKY\\_standard\\_parameters.txt](#): This text file contains the standard parameter. You can update them either directly in a text editor or using the Options menu in SPIKY. You can also create and use other files for different situations.

[SPIKY\\_default\\_standard\\_parameters.txt](#): This text file contains the default standard parameters. Please do not change it.

### **SPIKY-Logo**

[SPIKY-Logo.png](#): This file contains the SPIKY-Logo with white background.

[SPIKY-Logo\\_gray.png](#): This file contains the transparent SPIKY-Logo.