[4/26/12, BH: timestamp conversion](#h.31mt2t4q9qeg)

[5/3/12, BH: update findanalysis.m](#h.d2fexak9r6ot)

[5/3/12, BH: update listtag.m and findcellpos.m](#h.re9smmx0gl3q)

[5/3/12, BH: rewrite insertdata.m](#h.fkw2kgbk5jnb)

[5/3/12, BH: TheMatrix is a cell!](#h.f2ibj6nvvxe3)

[5/3/12, BH: listcell.m cleanup](#h.bzr9imr6j3ne)

[5/4/12, BH: maintain compatibility (findprop.m, getvalue.m)](#h.k1spqfd1hs6p)

[5/4/12, BH: abs2reltimes.m and rel2abstimes.m](#h.3lxwrz1lrhl)

[5/4/12, BH: iscellid.m](#h.naaj4ymg1vnh)

[5/7/12, BH: set\_subplots.m](#h.kahflnvmrog2)

[5/7/12, BH: plotwaveforms.m](#h.ubte9b8g4liq)

[5/8/12, BH: update extractSegSpikes.m](#h.aau1x5apyprx)

[5/8/12, BH, SPR: plot\_mclust\_projections2.m](#h.q0j8du3d8q8k)

[5/9/12, BH: check\_channel\_validity.m](#h.c1xolvoyldme)

[5/9/12, BH: lightcluster.m](#h.d2ngxq2plv1r)

[5/10/12, BH: prealignSpikes.m](#h.8bjt5l2230ie)

[5/10/12, BH: rel2abstimes.m](#h.up79fk479nkx)

[6/21/12, BH: new option for prealignSpikes.m: replace an event](#h.txycmb9nkksp)

[6/21/12, BH: delanalysis.m cleanup](#h.jh1juhfm1h33)

[6/26/12, BH: fixing addcell.m](#h.flxh7xahuy7c)

[7/5/12.BH: ultimate\_psth.m](#h.ypqa97o5updj)

[7/5/12, BH: selecttrial.m refurbished, added as option to filterTrials.m](#h.fmlrrv8clct)

[7/6/12, BH: addnewsessions obsolete](#h.cs87bfay0za5)

[7/6/12, BH: cellid2vals.m](#h.wk1604hkcm1j)

[7/6/12, BH: loadcb.m, findcell.m, selectcell.m cleanup](#h.8zzzg2bzlmul)

[7/6/12, BH: defineEventsEpochs\_default, defineEventsEpochs\_agonogo, defineEventsEpochs\_pulseon](#h.46n5gu4ovzku)

[7/6/12, BH: checkcb.m, checkcb2.m](#h.x7h6n81p01zr)

[8/3/12, BH: ultimate\_psth.m upgrade](#h.v8kp5g97vb0u)

[8/12/12, BH: ultimate\_psth.m upgrade #2](#h.pkwxmxf2rhye)

[8/21/12, BH: new functionality for selectcell.m](#h.l3b2zo3hzs6f)

[9/4/12, BH: partition\_trials.m fix](#h.bqq49lu85vwb)

[9/25/12, BH: parseTTLs.m](#h.6lttin89qs11)

[10/25/12, BH: findSegs2.m bugfix](#h.d20rdp3n0ix0)

[11/2/12, BH: addanalysis.m and delanalysis.m saves backup](#h.gvlhd4vfo4l7)

[11/2/12, BH: reorganizing, documentation](#h.drm7z59lnje9)

[11/7/12, SR: viewpopselectivity2a.m plot analyses from multiple cell groups.](#h.7ti8o4m2exnk)

[11/7/12, SR: setmyplot.m: added functionality.](#h.db8mt98sct78)

[11/9/12, BH: findSegs3.m](#h.1j07j68if8d2)

[12/6/12, BH: addanalysis bug fix](#h.3op3y7mzie3l)

[1/28/13, BH: helper functions](#h.kvwj3muczqa4)

[3/28/13, BH: formatforExcel](#h.hcmj1b8u3nmw)

[4/2/13, BH: MakeStimEvents2](#h.9hrlddarksp7)

[4/15/13, BH: plot functions](#h.bbn6ajpx3fev)

[4/16/13, BH: setvalue.m](#h.dx9lvtxd9px1)

[4/23/13, BH: addnewcells.m improved](#h.3dvfsm2mvtv7)

[5/16/13, BH: addanalysis.m](#h.642dub9zh523)

[5/20/13. BH: ultimate\_psth.m](#h.upue33nq3f55)

[5/20/13, BH: addanysis, addcell upgrade](#h.6cqsivmpnd5z)

[7/15/13, BH: findstimperiod.m](#h.y5cykclw4219)

[7/15/13, BH: psth\_stats.m](#h.j73hf71xx9dt)

[7/18/13, BH: formatforExcel.m](#h.kp5ldhjmxwq6)

# 4/26/12, BH: timestamp conversion

time stamp of TT... mat files will reflect the original Ntt file

timestamps are converted in loadcb every time the TT...mat file is loaded

no other conversion is needed, unless the original Ntt files are loaded

behavior timestamps remain in seconds

no need to edit MClust

# 5/3/12, BH: update findanalysis.m

two output arguments: first=position in TheMatrix (column index),

second=index in ANALYSIS

# 5/3/12, BH: update listtag.m and findcellpos.m

do justice to mice: now they accept ‘animal’ as an alternative of ‘rat’

# 5/3/12, BH: rewrite insertdata.m

kept former functionalities for compatibility

flexible calling, input handling with inputParser

analysis results can also be inserted, not only properties

can overwrite previous data

saves a backup before writing new cellbase file

documentation of the new version:

%INSERTDATA Inserts data into CellBase directly.

% NF = INSERTDATA(DATA,PARAM,VALUE) inserts DATA into TheMatrix

% and ANALYSES (see CellBase documentation). It optionally returns the

% list of entries in DATA which did not provide a match with CELLIDLIST

% (NF).

% Input arguments:

% DATA - Either a Matlab variable or a full pathname to a .mat or an

% Excel file. It can optionally contain a header row with

% 'property\_names' in the first column and corresponding property

% names for data columns. First column should contain cellIDs. If

% property names are added as both input argument and data

% header, the user is forced to choose which one to use.

% 'TYPE', T - Parameter indicating whether an analysis or a set of

% properties are added. T should start with the letter 'a' or 'p'

% accordingly.

% 'NAME', NM - Name of the analysis or properti(es). For analysis:

% it should be the name of a valid m file. For properties: either

% a character array (one property) or a cell array of strings

% (multiple properties).

% 'INPUT\_ARGUMENTS', LS1 - Input arguments for analysis, cell array.

% 'OUTPUT\_ARGUMENTS', LS2 - Names of output arguments for analysis,

% cell array of strings. This determines the property names when

% an analysis is added.

%

% The number of properties or output arguments (in case of adding values

% to an analysis) must match the number of data columns (number of all

% columns minus one) in DATA.

%

% Missing input arguments are asked for interactively. If DATA is empty,

% the user has to browse for the data file. If 'Cancel' is selected for

% browsing result, data values will interactively be prompted for

% according to a selected tag type (animal, session, tetrode or cell).

%

% If a property or an analysis is already added to CellBase, the user is

% asked whether to overwrite previous data. An analysis is only

% considered the same if the name of the m file and both the list of

% input and output arguments are the same. If the data is overwritten,

% the timestamp property in ANALYSES is updated.

%

% New data is inserted into TheMatrix and ANALYSES according to the

% corresponding cellIDs. A backup of the previous cellbase file is stored

% before overwriting.

%

% Examples:

% insertdata(data,'type','analysis','name','LRatio','input\_arguments',...

% {{'Amplitude','Energy'}},'output\_arguments',{'ID\_amp','Lr\_amp'})

% not\_found = insertdata([],'type','prop','name','validity');

%

% See also ADDANALYSIS and RUNANALYSIS.

# 5/3/12, BH: TheMatrix is a cell!

insertdata is compatible with both the cell and the double version; for double, new data columns are initialized with NaNs to avoid filling with zeros (cells pad with NaN by default!)

addanalysis: modified to deal with both cases; real test is going to be when I add a new cell to the CellBase

addcell: same

what’s left: delcell, delanalysis, exportdata, getvalue

listcell works, but output is now a cell - may give some problems, but it’s only called from example\_analysis and new\_example\_analysis1

# 5/3/12, BH: listcell.m cleanup

help added, variables initialized, comments added, ‘smart indent’

# 5/4/12, BH: maintain compatibility (findprop.m, getvalue.m)

edit findprop.m and getvalue.m to maintain compatibility with updated findanalysis.m and cell version of TheMatrix

findprop cleanup, helps and comments added, loads only ANALYSES from CellBase (loading TheMatrix is slow and unnecessary in this case)

getvalue: similar to runanalysis, returns double if conversion from cell is possible

# 5/4/12, BH: abs2reltimes.m and rel2abstimes.m

conversion between absolute time stamps and time stamps relative to an event

# 5/4/12, BH: iscellid.m

search in CELLIDLIST, return logical; calls validcellid with ‘list’ option

# 5/7/12, BH: set\_subplots.m

cleanup, comments, help

make it possible to call with axis property, value pairs

# 5/7/12, BH: plotwaveforms.m

function for plotting waveforms created:

%PLOTWAVEFORMS Plot spontaneous and/or light-evoked spike shape.

% PLOTWAVEFORMS(CELLID) plots the waveform of spontaneous and putative

% light-evoked spikes. It finds the latency of putative light-evoked

% spikes as peaks of firing rate after light pulses. The period for

% putative light-evoked spikes is defined by half-hight crossings around

% the peak (see FINDSTIMPERIOD for details). If there is no detectable

% activation, 1 and 5 ms are used, which include typical latencies of

% light-evoked spikes. Spontaneous spikes are extracted from 2 s periods

% before light bursts. Only 2000 randomly chosen spikes are plotted. This

% number can be changed using the 'MAXNUM', VALUE optional input

% parameter-value pair.

%

% Two optional input parameter value pairs control whether to plot both

% spontaneous and evoked waveforms or only one of those:

% HS = PLOTWAVEFORMS(CELLID,'SPONT',TRUE,'EVOKED',TRUE) plots both

% waveforms and a third plot to compare them. Setting either of the input

% values to 'false' will prevent plotting that type. Handles of the

% resulting figures are returned in the output structure HS with the

% following fieldnames: HS.H\_spont for spontaneous waveforms, HS.H\_evoked

% for light-evoked waveforms and HS.H\_compare for the compound figure.

%

% HS = PLOTWAVEFORMS(CELLID,'CORRELATION','TRUE') also returns waveform

% correlation for the largest channel (see NBSPIKESHAPECORR) in HS.R

% field of the output structure.

%

% Optional input parameter-value paris with default values:

% 'spont', true - plot spontaneous waveform

% 'evoked', true - plot light-evoked waveform

% 'maxnum', 2000 - maximum number of spikes to plot

% 'correlation', false - return waveform correlation

%

% See also FINDISSTIMPERIOD, ABS2RELTIMES, EXTRACTSEGSPIKES,

% EXTRACTSPIKEWAVEFORMS and NBSPIKESHAPECORR.

# 5/8/12, BH: update extractSegSpikes.m

overload with numeric input as first input argument: spike times (e.g. all spikes on a tetrode) can be passed

uses inputParser

# 5/8/12, BH, SPR: plot\_mclust\_projections2.m

plot projections, tagged cluster in orange, light-evoked spikes superimposed in blue

modified from plot\_mclust\_projections

%PLOT\_MCLUST\_PROJECTIONS2 Plot feature data for clusters.

% PLOT\_MCLUST\_PROJECTIONS2(CELLID) plots feature data for a tagged cell

% (CELLID) and it's tetrode pairs in all projections. Tagged cluster is

% shown in orange and light-evoked spikes are overlayed in blue. Time

% period for light-evoked activity is selected automatically (see

% FINDSTIMPERIOD). Only spikes from the beginning of the first to the end

% of the last stimulation protocol are included. These default behaviors

% can be modified using the following optional input arguments

% (parameter, value pairs, with default values):

% 'stim\_period', [] - start and end of stimulation period after each

% light pulse

% 'feature\_names', 'Energy' - features for which feature data are

% plotted; character or cell array

% 'marker\_size', 1 - marker size for the scatter plots

% 'usefastplot', true - use fast plotting method (downsample points

% to plot only one point per pixel; appears the same); faster,

% but zoom is not implemented (for saving in image formats, e.g.

% pdf or jpg); if false, full data is plotted (for viewing or

% saving fig format)

% 'stimonly', true - only spikes from the beginning of the first to

% the end of the last stimulation protocol are selected for

% plotting; if false, all spikes are included

% 'plotlightspikes', true - if true, light-evoked spikes are

% superimposed

%

% See also PLOTWAVEFORMS.

# 5/9/12, BH: check\_channel\_validity.m

outsourced from LRatio.m to form a separate function

%CHECK\_CHANNEL\_VALIDITY Check tetrode channels.

% VALID\_CHANNELS = CHECK\_CHANNEL\_VALIDITY(CELLID) loads 'Energy' property

% for all spikes on tetrode of CELLID and returns a 1-by-4 logical array

% that is true for the tetrode channels that have a mean Energy higher

% than an empirically decided threshold value.

%

% See also LRATIO.

# 5/9/12, BH: lightcluster.m

quantifying whether light-spikes are continuous with noise

%LIGHTCLUSTER Clustering of light-evoked spikes.

% [D L] = LIGHTCLUSTER(CELLID) analyzes whether the light-evoked spikes

% of a putative tagged cell (CELLID) are separated from other

% light-evoked spikes. It returns cluster quality indices (Isolation

% Distance, D; L-ratio, L) for the distance of light-evoked spikes of

% CELLID compared with all light-evoked spikes (i.e. cluster quality

% indices restricted to light-evoked spikes).

%

% [D L PREF] = LIGHTCLUSTER(CELLID) returns preferential occurrance of

% light-evoked spikes in the tagged cluster (PREF).

%

% [D L PREF VALID\_CHANNELS] = LIGHTCLUSTER(CELLID) also returns channel

% validity.

%

% See also LRATIO.

# 5/10/12, BH: prealignSpikes.m

option to use with a custom events or epochs matrix and not a definition file fixed;

input argument parsing with inputParser

ifsave option fixed

cleanup, help, comments added

# 5/10/12, BH: rel2abstimes.m

new option: call it with a cell, converts StimSpikes to abs time stamps

# 6/21/12, BH: new option for prealignSpikes.m: replace an event

an event can now be replaced with new data without deleting or recalculating previously added events

a new input parameter called writing\_behavior controls the now three options: overwrite the entire file, append to the file or replace preexisting events with new data

ifappend input param. is kept for backwards compatibility, but writing\_behavior takes precedence if specified

# 6/21/12, BH: delanalysis.m cleanup

help added, comments added, indented

# 6/26/12, BH: fixing addcell.m

addcell.m was bug fixed so that it can add analyses to TheMatrix when adding a new cell

addnewcell.m and iscellid.m modified so that the new cell is also detected as a valid cellID

# 7/5/12.BH: ultimate\_psth.m

new highlevel PSTH function with parseInput, many calling options

calls adaptive or normal PSTH

modular, fully cellbase-compatible

documentation:

%ULTIMATE\_PSTH Peri-stimulus time histogram.

% [PSTH SPSTH SPSTH\_SE] = ULTIMATE\_PSTH(CELLID,EVENT\_TYPE,EVENT,WINDOW,VARARGIN)

% calculates peri-stimulus time histogram (PSTH) for the cell passed in

% CELLID. Smoothed PSTH (SPSTH) and SE of smoothing (SPSTH\_SE) are also

% returned.

%

% Mandatory input arguments:

% CELLID: defines the cell (see CellBase documentation)

% EVENT: the event to which the PSTH is aligned

% EVENT\_TYPE: the type of event, 'stim' or 'trial'

% WINDOW: window for calculation relative to the event in seconds

%

% Default behavior of ULTIMATE\_PSTH can be modified by using a set of

% paramter-value pairs as optional input parameters. The following

% parameters are implemented (with default values):

% 'dt', 0.001 - time resolution in seconds

% 'sigma', 0.02 - smoothing kernel for the smoothed PSTH, in seconds

% 'margin',[-0.01 0.01] margins for PSTH calculation to get rid of

% edge effect due to smoothing

% 'event\_filter', 'none' - filter light-stimulation trials; see

% FILTERTRIALS for implemented filter types

% 'filterinput',[] - some filters require additional input; see

% FILTERTRIALS for details

% 'maxtrialno', 5000 - maximal number of trials included; if ther are

% more valid trials, they are randomly down-sampled

% 'parts', 'all' - partitioning the set of trials; input to

% PARTITION\_TRIALS, see details therein (default, no

% partitioning)

% 'isadaptive, true - if false, classic PSTH algorithm is applied; if

% true, adaptive PSTH is calculated (see APSTH2)

%

% See also BINRASTER2PSTH, BINRASTER2APSTH, APSTH2, VIEWCELL2B,

% PARTITION\_TRIALS and FILTERTRIALS.

# 7/5/12, BH: selecttrial.m refurbished, added as option to filterTrials.m

selecttrial: cleanup, help, preallocation, bug fix (now works for elementary logical expression as well)

filterTrials: a new option for filter named ‘custom’ is added; filterinput is used for calling selecttrial within this ‘custom’ filter

# 7/6/12, BH: addnewsessions obsolete

eliminated along with dependents

# 7/6/12, BH: cellid2vals.m

cleanup, fixed: was based on some other naming convention, and datenum output never worked

# 7/6/12, BH: loadcb.m, findcell.m, selectcell.m cleanup

cleanup, comments and help added

# 7/6/12, BH: defineEventsEpochs\_default, defineEventsEpochs\_agonogo, defineEventsEpochs\_pulseon

documentation and cleanup

# 7/6/12, BH: checkcb.m, checkcb2.m

checkcb obsolete, deleted; checkcb2 renamed checkcb; help and comments added, cleaned

# 8/3/12, BH: ultimate\_psth.m upgrade

proudly introduce ‘doubly adaptive’ PSTH: window size is also adapting (stronger adaptation);   
functions: dapsth.m, binraster2dapsth.m

ultimate\_psth can optionally call the ‘doubly adaptive’ algorithm now

# 8/12/12, BH: ultimate\_psth.m upgrade #2

ultimate\_psth is now equipped with the option of testing for sign. firing rate changes after an event; psth\_stats.m does this job

# 8/21/12, BH: new functionality for selectcell.m

selectcell now supports logical operations on string properties, which exist since TheMatrix is now a cell

example:

selstr = ['"PV+"==1&"ID\_PC">20&"Lr\_PC"<0.15&' ...

'ismember("Area1",{''GP'',''GP/SI'',''SI'',''IC'',''RT/IC'',''EP'',''EA'',''EAC''})'];

PV = selectcell(selstr);

# 9/4/12, BH: partition\_trials.m fix

NaN is not considered a separate partition now

# 9/25/12, BH: parseTTLs.m

MakeTrialEvents\_anogo (synchronization code between behav. and Neuralynx) now parses TTLs into binary numbers to treat the channels separately, so that superimposed TTLs do not pose a problem

calls parseTTLs.m for this

# 10/25/12, BH: findSegs2.m bugfix

a bug which messed up including/excluding behavior segments for different input arguments has been fixed

# 11/2/12, BH: addanalysis.m and delanalysis.m saves backup

these functions now save a timestamped backup of CellBase before overwriting, similar to delcell.m

# 11/2/12, BH: reorganizing, documentation

full html documentation for the database functions

m files reorganized to folders and subfolders in a more systematic way

# 11/7/12, SR: viewpopselectivity2a.m plot analyses from multiple cell groups.

This function plots a scatter plot of 2 analyses. 2 groups of cellids can be color coded in this plot.

# 11/7/12, SR: setmyplot.m: added functionality.

This version can check for Tags such as ‘title’ and ‘label’ and specify their FontSizes (or any other property) independently. This is possible only when passing a set of text handles as varargin. Need to specify ‘Tag’ property for these handles in the code. For e.g. l(1) = xlabel(‘The appropriate X label’,’Tag’,’label’). This version is compatible with earlier versions.

# 11/9/12, BH: findSegs3.m

a new version of findSegs that can handle missing stimulus or trial events

# 12/6/12, BH: addanalysis bug fix

previously, analyses had to return output arguments in a cell

this cumbersome syntax previously was eliminated

now addanalysis.m is fixed to be compatible with regular syntax

nargvout.m probably became obsolete

# 1/28/13, BH: helper functions

help added, commenting, cleanup

# 3/28/13, BH: formatforExcel

new helper function

changes special numbers (like NaN) to strings for writing them to Excel

# 4/2/13, BH: MakeStimEvents2

major overhaul, protocol names as optional input arguments, inputParser, documentation

# 4/15/13, BH: plot functions

plot functions restructured

help files

documentation

# 4/16/13, BH: setvalue.m

New function!

setvalue(cellid,property,value) sets a value in TheMatrix

a timestamped backup of cellbase is saved before overwriting

help:

%SETVALUE Set value associated with a property.

% STATUS = SETVALUE(CELLID,PROP,VALUE) sets a property (PROP) to a

% specified value (VALUE) for a given cell (CELLID) in CellBase. If

% editing TheMatrix is sucessful, STATUS = true is returned.

%

% See also GETVALUE.

# 4/23/13, BH: addnewcells.m improved

as a new input option, the directory in which new cells are located can be specified

the call syntax has been changed and inputParser is used to parse input arguments

findallcells.m has been modified to allow the new option

# 5/16/13, BH: addanalysis.m

bugs fixed, cleaned, changed input argument handling to inputParser

# 5/20/13. BH: ultimate\_psth.m

dynamic event definition added:

EVENT: the event to which the PSTH is aligned; if EVENT is a cell

% array of two strings, the first event is used for the PSTH

% and binraster before 0 and the second event is used for the

% PSTH and binraster after 0; if EVENT is a function handle, the

% function is called for CELLID to define the aligning event

% (dynamic event definition)

# 5/20/13, BH: addanysis, addcell upgrade

MAJOR UPGRADE: addanalysis and addcell now handles analysis functions that has mandatory input arguments, not only parameter-value pairs; also, it is not a requirement any more that all output arguments be added to TheMatrix; an optional parameter-value pair to addanalysis controls which subset of output arguments are inserted

%ADDANALYSIS Add an analysis to CellBase.

% ADDANALYSIS(FUNHANDLE) adds an analysis to CellBase with the specified

% function handle. ADDANALYSIS also executes the analysis function passed

% by FUNHANDLE on the cells included in CellBase. Note that the analysis

% specified by FUNHANDLE should take a cell ID as its first input

% argument.

%

% ADDANALYSIS(FUNHANDLE,'PROPERTY\_NAMES',PR) passes the list of property

% names (PR, cell array of strings) to ADDANALYSIS. The property name

% 'default' is used if no property names are specified. The first N

% outputs of the analysis function are used, where N is the number of

% inserted properties (length of PR).

%

% ADDANALYSIS(FUNHANDLE,'PROPERTY\_NAMES',PR,'OUTPUT\_SUBSET',OS) passes

% the list of property names (PR, cell array of strings) to ADDANALYSIS.

% The additional input argument OS determines which outputs of the

% analysis function are included. OS should be a numerical array indexing

% into the output argument list of the analysis function (integers

% between 1 and the number of FUNHANDLE output arguments - see NARGOUT).

% The length of OS should match the number of inserted properties (i.e.

% the length of PR).

%

% ADDANALYSIS(FUNHANDLE,'PROPERTY\_NAMES',PR,'OUTPUT\_SUBSET',OS,'MANDATORY',ARGM,'ARGLIST',ARGVAL)

% also passes input arguments to FUNHANDLE. Mandatory arguments should be

% passed in ARGM and parameter-value pairs should be specified in ARGVAL

% (N-by-2 cell array). These are be passed on to the analysis function.

%

% Examples:

% addanalysis(@LRatio2,'property\_names',{'ID\_PC','Lr\_PC'},'arglist',{'fea

% ture\_names' {'WavePC1' 'Energy'}})

%

% addanalysis(@ultimate\_psth,...

% 'mandatory',{'trial' @findAlignEvent\_negfeedback\_gonogo [-0.6 0.6]},...

% 'property\_names',{'FA\_psth' 'FA\_psth\_stats'},'output\_subset',[1 6],...

% 'arglist',{'dt',0.001;'display',false;'sigma',0.02;'parts','all';'isadaptive',2;...

% 'event\_filter','custom';'filterinput','FalseAlarm==1';'maxtrialno',Inf;...

% 'baselinewin',[-0.5 0];'testwin',[0 0.5];'relative\_threshold',0.1});

%

% See also FINDANALYSIS.

# 7/15/13, BH: findstimperiod.m

updated with input parser and renamed findStimPeriod

bug fixed: activation time returned in sec now (did not affect other programs, this output was never used)

minor change: margin excluded from baseline prob. (should have at max negliable effects on nbtaggedprop)

however, the following call makes findStimPeriod obsolete:

[psth, spsth, ~, ~, spt, stats] = ultimate\_psth(cellid,'stim',g.event,[-0.005 0.01],...

'dt',0.0005,'display',false,'isadaptive',1,'maxtrialno',Inf,...

'event\_filter',g.event\_filter,'filterinput',g.filterinput,...

'baselinewin',[-0.005 0],'testwin',[0 0.01],'relative\_threshold',0.5,'margin',[0 0]);

lim1 = stats.activation\_start; % window start for evoked spikes

lim2 = stats.activation\_end; % window end for evoked spikes

L = stats.activation\_peak; % LATENCY

at = stats.activation\_time;

M = stats.maxvalue;

B = stats.baseline;

- new syntax is slightly slower (on an example cell, 1.74 s vs 1.56 s)

- new syntax has more options, e.g. it is easy to use doubly adaptive PSTH

# 7/15/13, BH: psth\_stats.m

error handling; warning if MW-test skipped due to short baseline

changed window definitions to exlculde first point from baseline and last from test; this way it’s easy to avoid indexing issues

should have only negligible effects on nbtaggedprop

# 7/18/13, BH: formatforExcel.m

bug fixing

replacement of Infs and NaNs was crashing on empty elements of TheMatrix