KING'S College LONDON

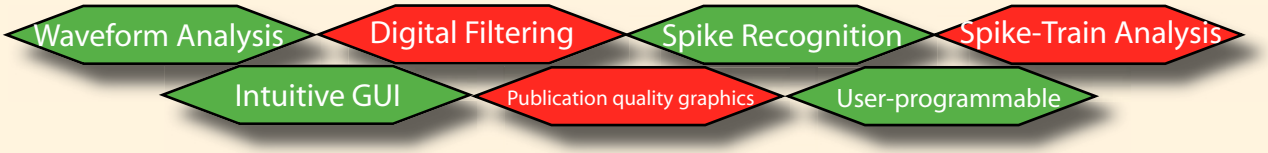
http://sigtool.sourceforge.net/

Version 0.93

sigTOOL

Open-source Freeware

sigTOOL has been developed by Malcolm Lidierth in the Wolfson Centre for Age-Related Diseases at the Guy's Hospital Campus of King's College London, London UK.



Developed to run within MATLAB [1], sigTOOL provides a programming and analysis environment for processing neuroscience data. A graphical-user interface to this environment provides the user with a self-contained application for waveform and spike-train analysis. User-written extensions to this application can be added to the interface on-the-fly without the need to modify any of the existing code.

sigTOOL is an open-source package distributed under the GNU General Public Licence.

A short paper describing sigTOOL is available: M.Lidierth (2009). sigTOOL: a MATLAB-based environment for sharing laboratory-developed software to analyze biological signals. Journal of Neuroscience Methods 178, 188-196 http://dx.doi.org/10.1016/j.jneumeth.2008.11.004

DATA IMPORT

sigTOOL provides the tools to import data into MATLAB from many commonly used proprietory data files. These include:

		Windows 32 bit	Windows 64 bit	Linux 32 bit	Linux 64 bit	Intel Mac 32 bit	Intel Mac 64 bit	Pre Intel Mac
ABF[2]	Molecular Devices Inc e.g. pClamp, AxoScope, Clamp-Fit software (ABF v10 and earlier).	●	●					
CFS[2]	Cambridge Electronic Design Ltd– Signal software	●	●					
DAT	HEKA - PatchMaster/Chart-Master	●	●	●	●	●	●	●
MAP[1]	Alpha Omega	●	●					
MCD[1]	Multi Channel Systems	●	●	●	●	●	●	
NEX[1]	Nex Technologies – NeuroExplorer software	●	●					
NEV[1]	BlackRock Systems (Cyberkinetics Inc)	●	●	●				
NSN[1]	Neuroshare native	●	●					
PLX[1]	Plexon Instruments	●	●					
SMR/ SON	Cambridge Electronic Design Ltd– Spike2 software	●	●	●	●	●	●	●
SSD/ DAT	University College London CONSAM data	●	●	●	●	●	●	●
STAM	Weill Medical College (STA Toolkit)	●	●					
WAV	Audio file	●	●	●	●	●	●	●
MPG/AVI etc	Multimedia files[3]	●	●	●	●	●	●	●

Key

● Supported and tested

● Probably supported but not yet tested on the target platform

● Untested. User may need to recompile some code files for the target platform

[1] Supported via NeuroShare MATLAB API and manufacturer's shared library

[2] Supported via manufacturer's Windows DLL

[3] Supported via Micah Richert's mmread function [8]

Unlisted software may be supported where export of data to one of the file formats above is available, e.g. Bruxton Acquire and AD Instruments LabChart via ABF , AD Instruments Chart via WAV.

DATA ANALYSIS

Many standard analysis methods have been built into sigTOOL. These provide facilities for analyzing waveform and spike-train data and presently include:

WAVEFORM	SPIKE TRAINS
Averaging (mean and median),	Interspike interval distributions
Auto- and cross-correlation,	Poincaré plots
Power spectral analysis,	Stimulus driven peri-event time histograms
Coherence estimation,	Rasters and frequencygrams
Digital filtering (IIR and FIR),	Event auto- and cross- correlations
Resampling	Joint peri-stimulus time histograms
Amplitude distribution.	Spike-triggered averaging.
Independent component analysis (ICA) via FastICA and Icasso [3].	Automatic spike recognition is supported through an interface to the Waveclus spike sorting package [4]

DATA STORAGE

Imported data are represented in the standard MATLAB (MAT) file format and memory mapped from disc to the system virtual memory. This allows very large data sets to be represented as 'virtual' matrices in the MATLAB environment and accessed using standard MATLAB syntax. sigTOOL manages the data dynamically to optimize performance.

Within sigTOOL, data has been organized on disc, and in PC memory, to maximize speed. and to take full advantage of MATLAB's speed-optimized matrix manipulation routines. This organization also increases the likelihood that computations will be performed in CPU cache memory, offering a further ~10x speed improvement [5].

DATA TYPES

sigTOOL supports waveform data that has been sampled continuously or episodically. Multiplexed channels (e.g. from tetrode recordings) are also supported together with event data (e.g. TTL pulses or levels). Each event or waveform epoch can be associated with a set of marker values (e.g. a set of numeric or character values). The marker values can also be replaced with more complex metadata for use in neuroinformatics databases. A single common structure has been developed to support all of these data types. This highly versatile structure can readily be used to support other data formats e.g. image data from a video recording can be incorporated directly into a sigTOOL data file together with electrophysiological data.

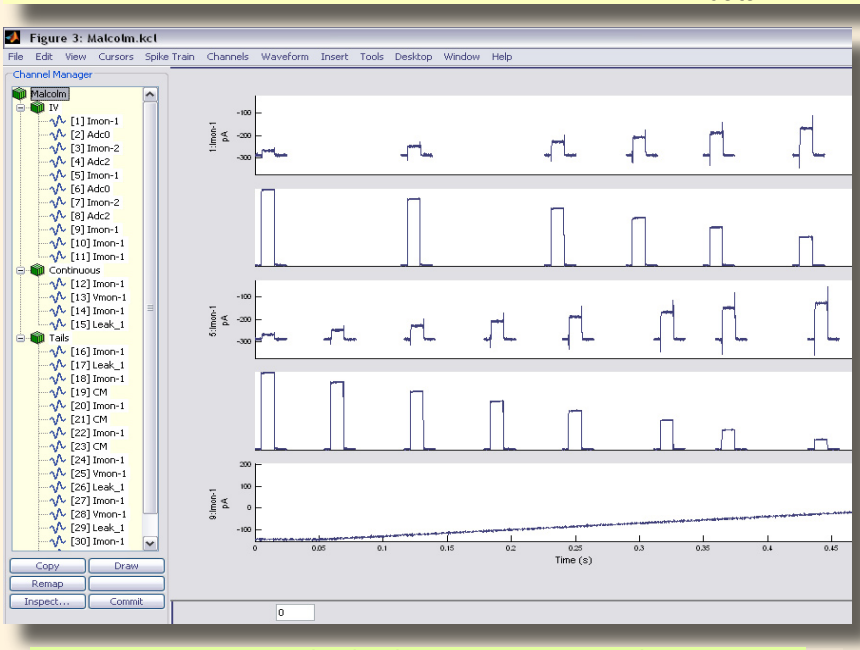


Figure 1: Data visualized in the sigTOOL GUI on Windows XP.

DATA VISUALIZATION

sigTOOL provides a strip-chart style data display for the data. This is embedded in an interactive graphical user interface, the sigTOOL GUI, that provides access to the sigTOOL data analysis routines through its menus. The Channel Manager allows selection of the channels to display and selected channels can be drag-and-dropped into the analysis menus

THE sigTOOL GUI

The sigTOOL graphical-user interface provides access to all of the data import, analysis and export facilities. The user needs no knowledge of sigTOOL or of MATLAB to use this interface. The GUI also provides mechanisms for batch processing multiple files and has a 'History' function to record user commands.

VISUALIZING SINGLE DATA CHANNELS

sigTOOL provides facilities to examine single data channels in a high-speed scrollable window. The data display can be controlled interactively through the GUI and programmatically by user-developed code.

CUSTOM CODE DEVELOPMENT

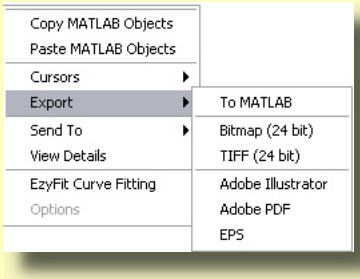
sigTOOL was designed primarily as a code development tool to assist users to develop novel analysis algorithms and to make these portable between operating systems, file formats and laboratories.

sigTOOL is heavily object-oriented: data channels and analysis results are both represented as MATLAB objects. The methods for the channel objects provide an easy interface to the data stored on disc while those for result objects automatically provide access to the standard sigTOOL data export functions.

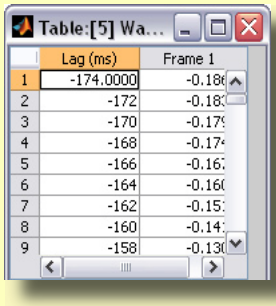
The sigTOOL GUI is self-modifying and will automatically incorporate user-developed routines. Full sigTOOL functionality will be available to support those routines from the GUI, including data and graphics export. The History recording function also automatically supports user-developed code.

EXPORTING RESULTS

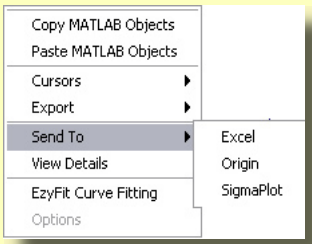
Graphical results can be exported in a number of bitmapped (BMP, TIF) or vector formats (PDF, EPS and Adobe Illustrator) simply by clicking on the display.



Results may also be viewed as numerical values in an internal sigTOOL spreadsheet allowing cut & paste to external analysis software.



On Windows platforms, direct output of results to external software is supported via ActiveX (to Microsoft Excel, Systat SigmaPlot and OriginLabs Origin software).



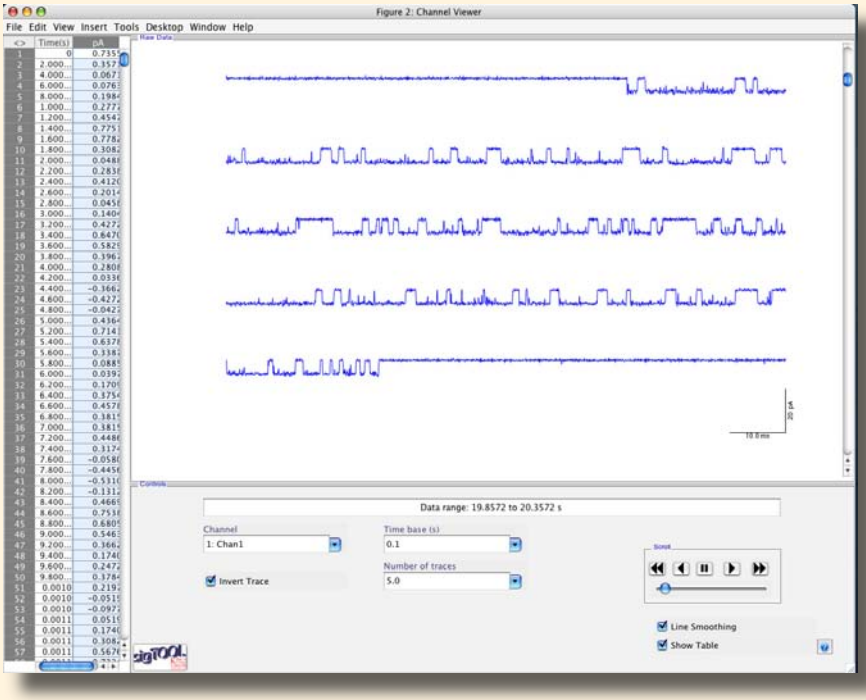


Figure 2: Data visualized in the sigTOOL GUI Channel Viewer on a Apple Mac PowerPC here showing a single ion channel recording form a wild type mouse ACh receptor courtesy of Dr Remigijus Lape at University College London

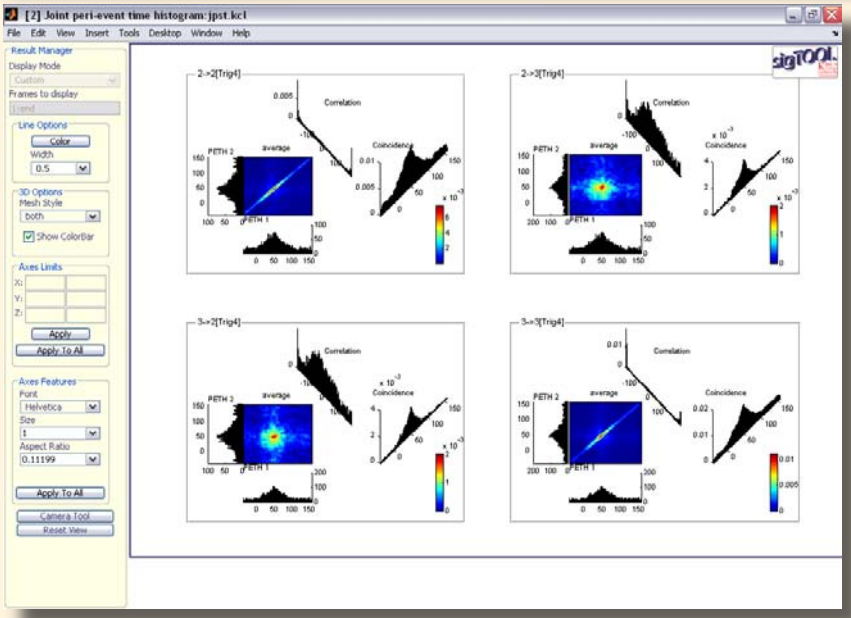


Figure 3: Joint peri-stimulus time histograms. Spike trains are test data from the Mulab website at the University of Pennsylvania http://mulab.physiol.upenn.edu/

RECORDING AN ANALYSIS HISTORY

The sigTOOL GUI provides an option to record a history of user actions. The history is written to a standard MATLAB code file (an m-file) and may be used to batch process sigTOOL data files from the GUI.

The history feature provides automatic support for user-developed code as well as the standard sigTOOL analysis functions.

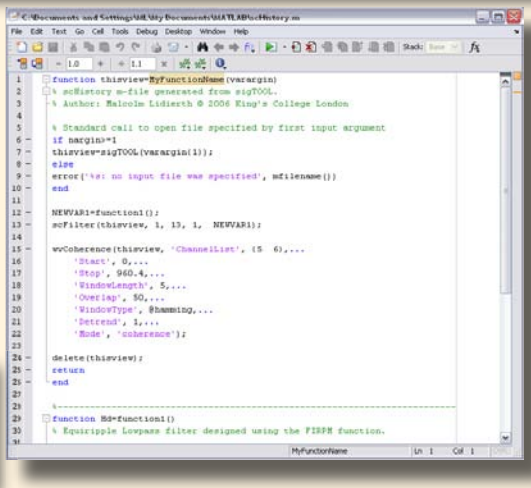


Figure 5. Example of a history file generated in sigTOOL and viewed in the standard MATLAB code editor

Table summarizing the main features of sigTOOL

Standard sigTOOL facility	Advantage for end-user programmer	Advantage for end-user analyser
Commonly used laboratory file formats can easily be imported into the common sigTOOL data format	No knowledge required of the original data file format	Code developed can be applied to any supported file format and is therefore immediately portable between laboratories.
	Versatility. The sigTOOL data format supports spike trains, continuously and episodically sampled waveform data, video and image data.	Integrated facilities to analyze multiple data types
Raw data is represented within sigTOOL-defined objects	These objects use memory mapping from disc through the operating system's virtual memory.	Ability to analyze very large data files
	Access to the data is via the easy-to-use methods associated with those objects. No detailed knowledge is required of the sigTOOL data format	Speed. The methods have been heavily profiled and optimized.
	Code within the object methods is optimized for speed and memory management for both disc and RAM access. Data are organized to take full advantage of MATLAB's matrix algebra functions.	Speed. Data that are likely to be accessed in sequence are organized in contiguous memory areas. This speeds data access and optimizes the chances of calculations being performed in CPU cache memory (providing an additional speed enhancement of ~10 fold)[5].
	The object methods deal transparently with platform-specific data issues such as IEEE endian format	Platform independence. To-date, sigTOOL has been tested on Windows XP, Fedora Linux and Mac OS10.4
Results are represented within sigTOOL-defined objects	These objects are fully supported without further programming	The plot method invokes the standard sigTOOL result view providing tools for user interaction (selection of plot type, colour scheme, line thickness, axes limits etc).
	Support for printing, and exporting data and publication quality graphics is made available through the associated methods without further programming	Full sigTOOL functionality available including printing, and exporting both raw data and publication graphics. Spreadsheet access to data. ActiveX interaction with Excel, SigmaPlot and Origin software on Windows platforms.
sigTOOL GUI	User-written code is incorporated into the GUI automatically	Ease of adding third-party developed extensions
	Provides easy access to a full range of MATLAB built-in and Java Swing interfaces	User-friendly GUI
	Records a user history that includes support for third-party additions.	History files can be used for batch processing data files

REQUIREMENTS FOR sigTOOL

sigTOOL requires MATLAB R2006a (version 7.2) or later. Some functions require the Signal Processing, Statistics or Wavelet Toolboxes.

sigTOOL is largely platform-independent and has been tested on Windows XP, Linux and Mac OS10.4. File import functions may be restricted to one or more platforms.

REFERENCES

[1] http://www.mathworks.com/
[2] http://www.neuroshare.org/
[3] FastICA by Hugo Gävert, Jarmo Hurri, Jaakko Särelä and Aapo Hyvärinen.
Icasso by Johan Himberg.
http://www.cis.hut.fi/projects/ica/fastica/
[4] http://www.vis.caltech.edu/~rodri/Wave_clus/Wave_clus_home.htm
[5] http://www.mathworks.com/company/newsletters/news_notes/june07/patterns.html
[8] http://www.mathworks.com/matlabcentral/fileexchange/8028

Updated for sigTOOL 0.93 January 2010 (ML)