SWAN: A tool to track single units across consecutive electrophysiological recordings

Shashwat Sridhar[1], Alper Yegenoglu[1,2], Nicole Voges[1], Thomas Brochier[3], Alexa Riehle[3,1], Sonja Grün[1,4], Michael Denker[1]

[1] Institute of Neuroscience and Medicine (INM-6) and Institute for Advanced Simulation (IAS-6) and JARA Institute Brain Structure-Function Relationships (INM-10), Jülich Research Centre, Jülich, Germany

[2] Institute for Advanced Simulation, Jülich Supercomputing Centre (JSC), SimLab Neuroscience, Jülich Research Centre, Jülich, Germany

[3] Institut de Neurosciences de la Timone (INT), CNRS-Aix-Marseille University, UMR 7289, Marseille, France

[4] Theoretical Systems Neurobiology, RWTH Aachen University, Germany

Contact: s.sridhar@fz-juelich.de

Tracking Single Units

In electrophysiology, the process of spike sorting [1, 2] is applied to recorded time series to extract spike data and identify single unit activity (SUA). Usually, each session of an experiment is spike sorted independently, so the resulting units are assumed to be **independent** of each other.

However, with **chronically implanted** microelectrode arrays, the signal recorded at each electrode during multiple recording sessions is likely to originate from the same population of neurons. In such a case, one cannot assume that units detected on the same electrode on different days are independent of each other. This affects any **population statistics** drawn on the ensemble of all spike sorted units. A solution to this problem is to identify which units from different recording sessions correspond to the same neuron, i.e., to **track single units** across sessions [3, 4].

SWAN (Sequential Waveform Analyzer) is a tool specifically developed for this task. It:

- provides a **per-electrode** visualization of a large number of spike-sorted datasets extracted from multiple recording sessions
- is designed such that each group of similar units is assigned a unique **global unit id** (identified by a unique colour)
- organizes the user interface into **views**, each showing a different feature of the units being analyzed
- allows the reassignment of global unit ids to units through a **swapping** mechanism (see right)
- bundles an algorithm that **automatically** suggests an assignment of global unit ids

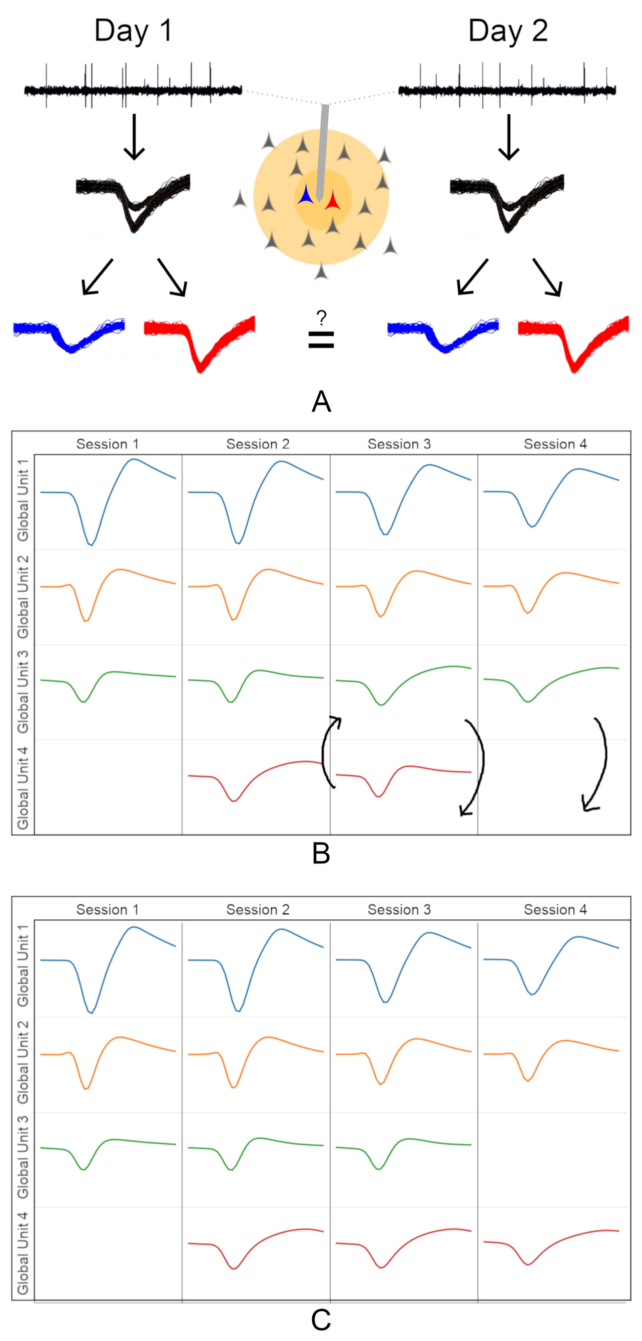

Fig. 1 *Single Unit Tracking.* **A** Panel modified from [8]. Unit tracking involves the comparison of spike-sorted units (shown in red and blue) from different recording sessions to determine whether they represent the same neuron in the brain. **B** MxN grid that mimics the main plot grid in SWAN (M - global unit ids, N - sessions). Each panel contains the mean waveform of the corresponding unit. By sequentially swapping units in columns (black arrows) the best mapping of units to global unit ids is obtained **C**.

SWAN Software Information

SWAN (Sequential Waveform Analyzer) is a tool written in Python to analyze the long-term stability of single units in spike-sorted datasets that were recorded chronically using microelectrode-arrays. It provides an intuitive graphical interface and tools to aid the user, along with implementations of algorithms from literature that automate this analysis.

Installation pip install git+https://github.com/INM-6/swan.git

The QR codes link to SWAN, Neo and Elephant, respectively. Neo and Elephant are Python packages for electrophysiological data storage and analysis, and are used extensively in SWAN.

Summary

- SWAN (Sequential Waveform Analyzer) - tool to identify putative identical neurons across multiple recording sessions in electrophysiological data from multi-electrode arrays and analyze their stability
- Modular and flexible GUI, organized in several views that display different aspects of the spike waveforms and the spike time statistics
- Mapping of units to global unit ids across sessions can be saved in the odML format as metadata [6, 7]
- Includes algorithm for automatic unit tracking
- Outlook:
  - implement existing automatic algorithms from literature [3, 4]
  - include support for non-Blackrock file formats

Automatic Unit Tracking

As an alternative to manually tracking units, SWAN currently includes two algorithms that automatically calculate a suggested final mapping of units to global ids, which can be be manually modified.

High-dimensional clustering algorithm

1. A feature vector is constructed for every unit. Features considered - mean waveform, first two derivatives of mean waveform, inter-spike interval histogram, CV2, LV, firing rate
2. Units are clustered in feature space using the KMeans++ algorithm
3. Clusters are curated:
   - Session-duplicates: units having session-conflicts with other units in cluster are assigned either moved to an existing cluster or are assigned a new cluster
   - Timestamp-split: clusters containing units with a time difference exceeding a user-defined threshold **d** are split into smaller clusters

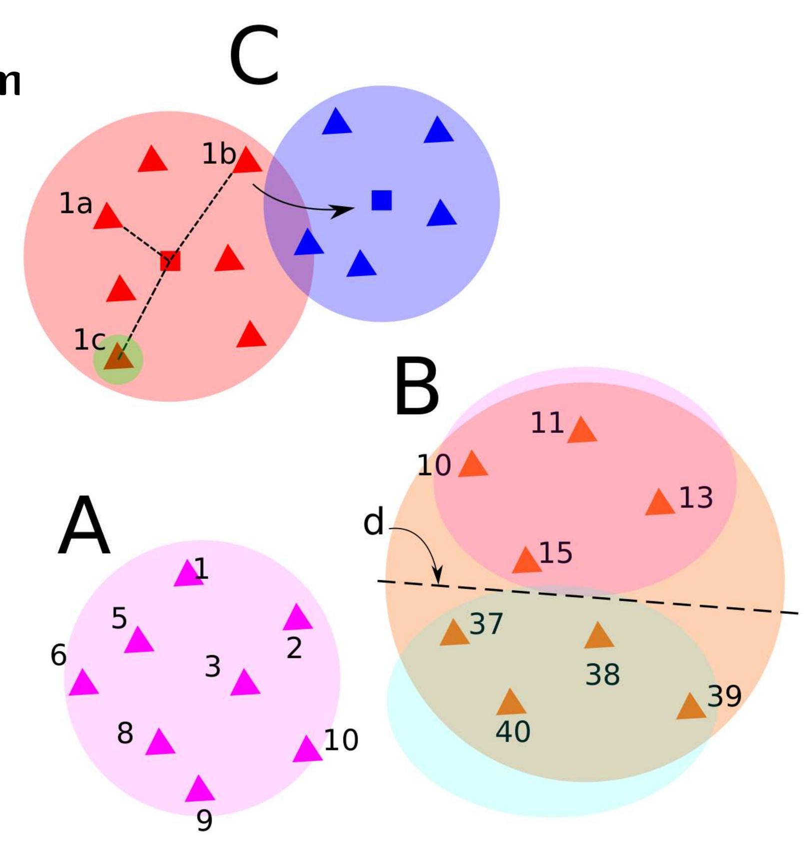

Fig. 2 *High-dimensional clustering algorithm.* Triangles - units, squares - cluster centers, circles - clusters, numbers - day of recording for unit. **A** a simple clustering result - units are nearby in feature-space and time. **B** a larger cluster (orange) being split into smaller clusters (cyan and pink) on the basis of timestamp-split according to the user-defined parameter *d*. **C** session-conflicts i.e. multiple units in the red cluster are from the same session (session 1). Of the three conflicting units, one is assigned to a nearby existing cluster (1b, red to blue). Another is assigned a new cluster (1c, red to green).

Euclidean distance-based algorithm

- All pairs of consecutive sessions are successively compared to determine similar units
- Two units are marked similar if the Euclidean distance between their mean waveforms is less than the average distance over all pairs of units in the two sessions
- Drawback: Units that disappear and reappear cannot be assigned the same global unit id

Illustrative Example of Unit Tracking Using SWAN

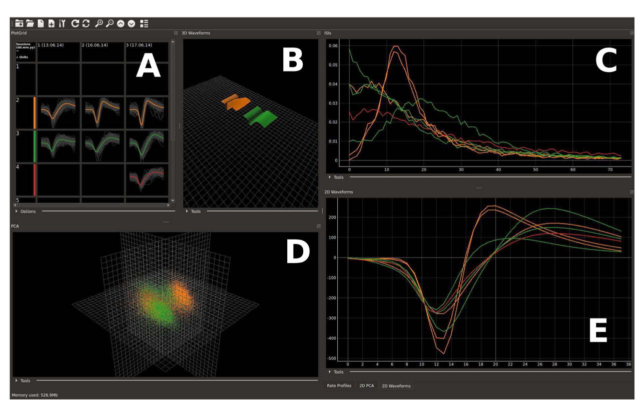

Fig. 3 *SWAN with three loaded sessions.* All units with the same global unit id, share the same color. **A** Grid of plots representing the current mapping of session unit ids to global unit ids. Global ids are represented as rows, recording sessions as columns; **B** Progression of the mean waveforms of units over successive recording sessions; **C** Inter-spike interval histograms; **D** Scatter plot of first three principal components after performing principal component analysis on all waveforms; **E** Mean waveforms of all units.

Goal: to identify groups of identical units in a sequence of 3 recording sessions on one channel from data recorded using a 96 channel Utah Array [5].

Step 1: Preparation
Select and load the three recording sessions. Activate and arrange the different views of the loaded waveform data (*Fig. 3*).

Step 2: Visually inspect mean waveforms
Identify units with similar mean waveforms in panels A and E by visual inspection. These units have been highlighted in *Fig. 4* with the same background and arrow colors, respectively. We observe two distinct groups of similar units and two independent units.

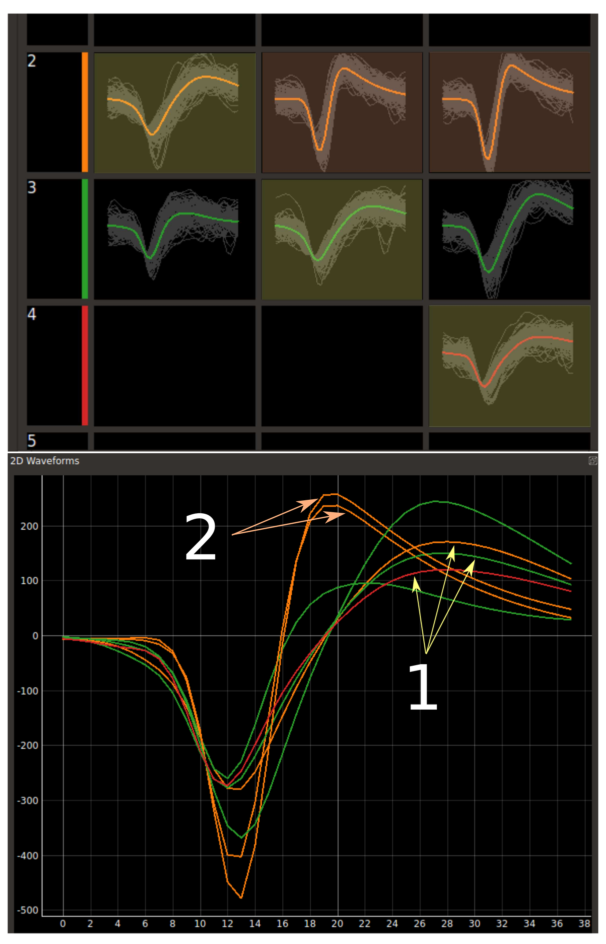

Fig. 4 *Plot grid (view A in Fig. 2) and mean waveforms (view C).*

Step 3: Visually inspect ISI histograms
Identify units with similar inter-spike interval histograms in panel C by visual inspection. Similar histograms have been labeled *1* and *2* in Fig. 5 for the two distinct groups, respectively.

Step 4: Confirm with PCA
Identify clusters (corresponding to units) which are close to each other in panel D. The closer they are in PC space, the more similar their waveform shapes. In the current example, the units identified as similar in steps 2 and 3 are also found to cluster in PC space.

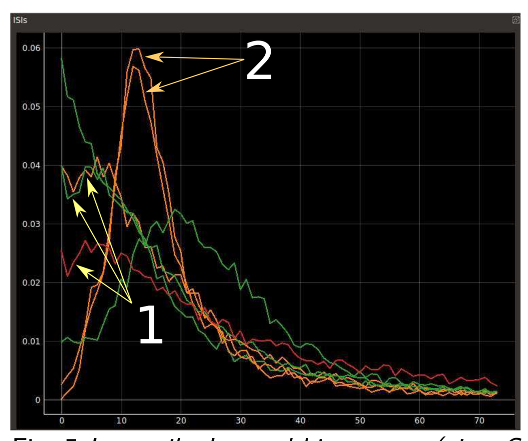

Fig. 5 *Interspike Interval histograms (view C in Fig. 3).*

Step 6: Automate
Steps 2 to 5 can be replaced by running the automatic algorithm (see box above). The algorithm yields a result similar to that obtained in step 5.

The final mapping of unit ids to global ids for all channels can be saved and exported for further analysis.

Step 5: Swap
Swap the positions of the units in the mapping in panel A so that each row contains only those units that are similar. Swaps can be performed by selecting the two desired cells in the plot grid and clicking on the *Swap* button on the top panel.

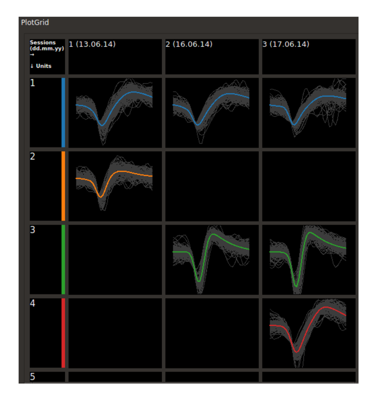

Fig. 6 *Final mapping at the end of the swapping step.* Group 1 of units has been moved to the blue row. and group 2 has been moved to the green row. The other two units have been assigned unique global ids.

References

[1] Einevoll, G. T. et al. Current Opinion in Neurobiology 22, 11–17 (2012).
[2] Buzsáki, G. Nature Neuroscience 7, 446–451 (2004).
[3] Dickey, A. S. et al. Journal of neurophysiology 102, 1331–1339 (2009).
[4] Fraser, G. W., Schwartz, A. B. Journal of Neurophysiology 107, 1970–1978 (2012).
[5] Brochier, T. et al. Scientific Data 5, 180055 (2018).
[6] Grewe, J., Wachtler, T., Benda, J. Frontiers in Neuroinformatics 5, 16 (2011).
[7] Zehl, L. et al. Frontiers in Neuroinformatics 10, 26 (2016).
[8] Quiroga, R. Q. (2007) Spike sorting. Scholarpedia, 2(12):3583.

Acknowledgments: EU Horizon 2020 Framework Programme for Research and Innovation under Specific Grant Agreements No. 720270 and No. 785907 (Human Brain Projects SGA1 and SGA2); DFG Grants DE 2175/2-1 and GR 1753/4-2 Priority Program (SPP 1665); International Associated Laboratory (LIA) “Vision for Action” between CNRS and Aix-Marseille Univ, Marseille, France, and Research Centre Juelich, Germany. We thank Christoph Gollan, who implemented a previous version of the SWAN tool.