Neurophysiological Signal Processing with MATLAB: Initial Steps - Data Loading, Filtering, and Spike Detection

A Comprehensive Guide to Identifying Neuronal Action Potentials from Noisy Recordings

Understanding Spike Sorting in Neuroscience Data Analysis

Spike sorting is a fundamental process in neuroscience data analysis, designed to extract action potentials, or "spikes," originating from individual neurons amidst noisy multi-unit extracellular recordings. The complexity of this task is compounded by the presence of spikes from multiple neurons and significant noise in the raw data captured by extracellular electrodes. The overall process of spike sorting involves several key steps:

Step 1: Load

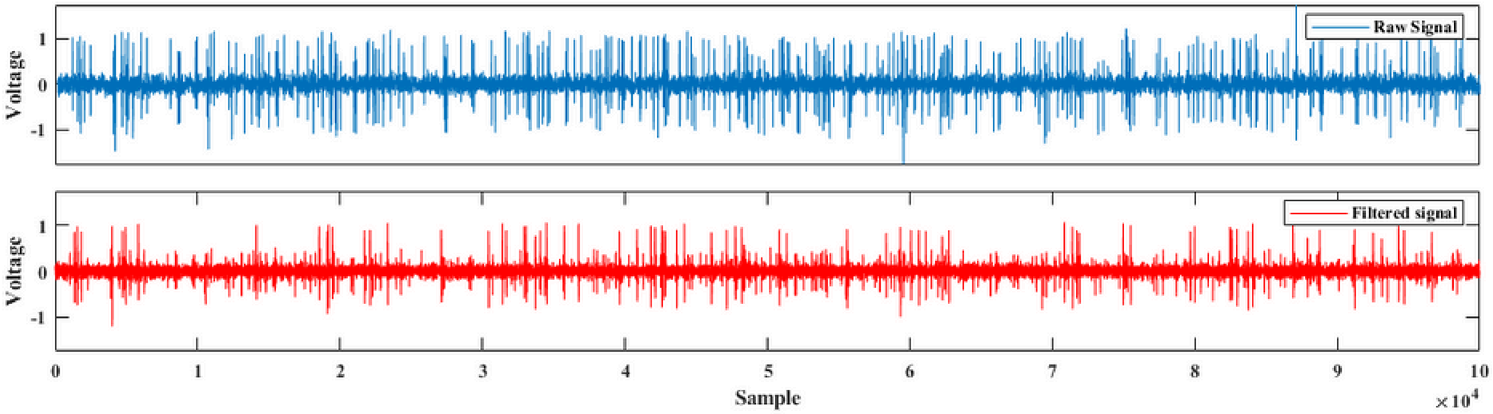
In this step, users can load data in various formats, including `.mat`, `.txt`, and `.excel`, providing flexibility for different data sources. The process begins with selecting the input file using the "Open File" option, setting parameters such as normalization, and choosing input channels. The loaded data is then prepared for further filtering and analysis.

Step 2: Filtering

This step involves applying filtering techniques to the loaded data to enhance signal quality. Users can select from different filter types, including Butterworth (butter), Chebyshev Type I (Cheby1), Chebyshev Type II (Cheby2), and Elliptic (Ellip), each offering distinct properties for signal processing. The software is designed to handle single-channel extracellular recordings of neural activity, which typically exhibit low-amplitude spikes embedded in noise, with frequency content often ranging from 300 Hz to 6 kHz.

- **Butterworth Filter**: Known for its flat frequency response in the passband, this filter is suitable for general-purpose filtering where minimal distortion is desired.
- **Chebyshev Type I Filter**: This filter provides a sharper roll-off compared to a Butterworth filter; however, it does introduce ripple in the passband. It's useful for emphasizing spike features when some ripple is acceptable.
- **Chebyshev Type II Filter**: Features ripple in the stopband, providing better attenuation of out-of-band noise, ideal for removing specific noise frequencies.
- **Elliptic Filter**: Combines passband and stopband ripple for the sharpest transition, making it ideal for isolating neural spikes with high selectivity, albeit with potential distortion.

For single-channel extracellular recordings, the Butterworth filter is often suitable due to its balanced performance and minimal phase distortion, preserving the temporal characteristics of neural spikes. Users can set parameters like frequency (Fs, Fpass, Fstop), and order to tailor the filtering process



Step 3: Spike Detection

Spike detection is a critical step in identifying the actual neural spikes within the filtered data. Users can choose the type of input signal for analysis:

1. **Input Options**:
   - **Raw Signal:** The unprocessed signal directly from the data source.
   - **Filtered Signal:** The signal post-filtering to reduce noise.
   - **1st Derivative of Filtered Data:** Emphasizes amplitude changes, enhancing spike detection by highlighting rapid transitions.
2. **Detection Type**:
   - **Positive Peaks:** Detects spikes where the signal amplitude rises above a threshold.
   - **Negative Peaks:** Detects spikes where the signal amplitude falls below a threshold.
   - **Both Peaks:** Detects both positive and negative peaks.

Threshold settings are crucial in spike detection:

- **Manual**: Users set the threshold (Thr_min, Thr_max) based on their observations and expertise with the dataset, ensuring precise distinction between spikes and noise.
- **Automatic**: The software calculates the threshold automatically, beneficial for users with less detailed dataset knowledge or when processing large datasets consistently.

Identifying Spike Indices

Once a threshold is applied, contiguous data blocks corresponding to individual spikes are detected. This is typically done by pinpointing indices marking the start of each threshold spike.

Define Part Set Events

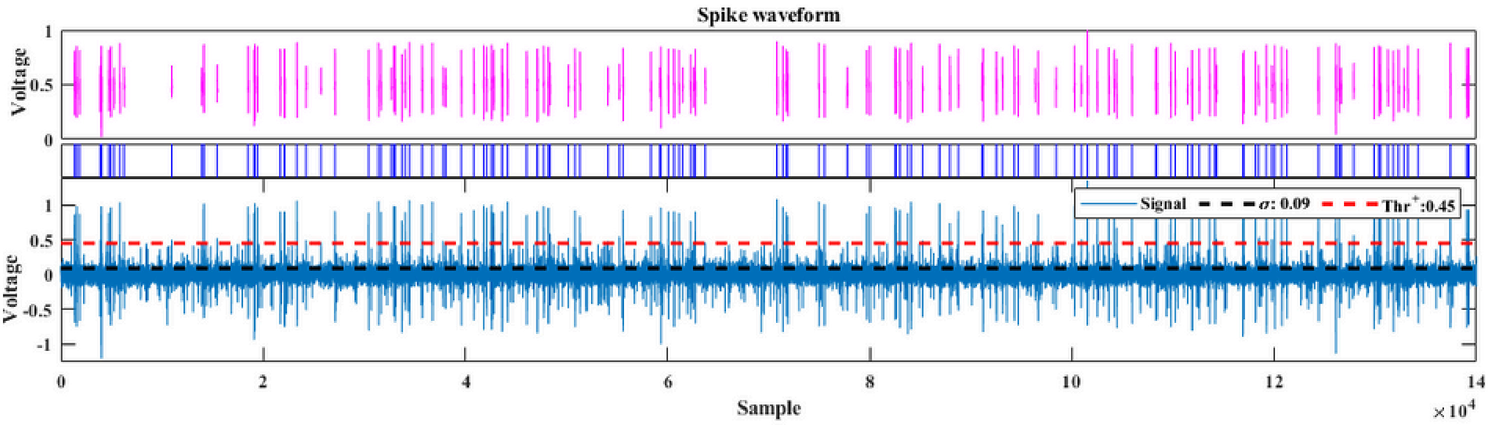
The "Set Events" section defines the temporal window around each detected spike for further analysis:

- **Number Pre-Event {+/-}:** Specifies the samples before the spike event to include in the analysis, capturing pre-spike context.
- **Number Post-Event {+/-}:** Specifies the samples after the spike event, capturing post-spike behavior.
- **Dead Time:** Sets a refractory period during which no additional spikes are detected, preventing multiple detections of the same event.

Extracting and Aligning Waveforms

Once candidate spike start-points are identified, a specific duration of voltage data is extracted to generate an array of spike waveforms. Each waveform consists of numerous data points representing the digitized voltage over a specific period. These waveforms are then aligned in a large array, excluding events with peaks above a certain voltage to filter out artifacts or superimposed spikes.

Adjustments to these values, based on sampling rate and signal characteristics, optimize spike extraction, ensuring accurate data analysis in neuroscience research.



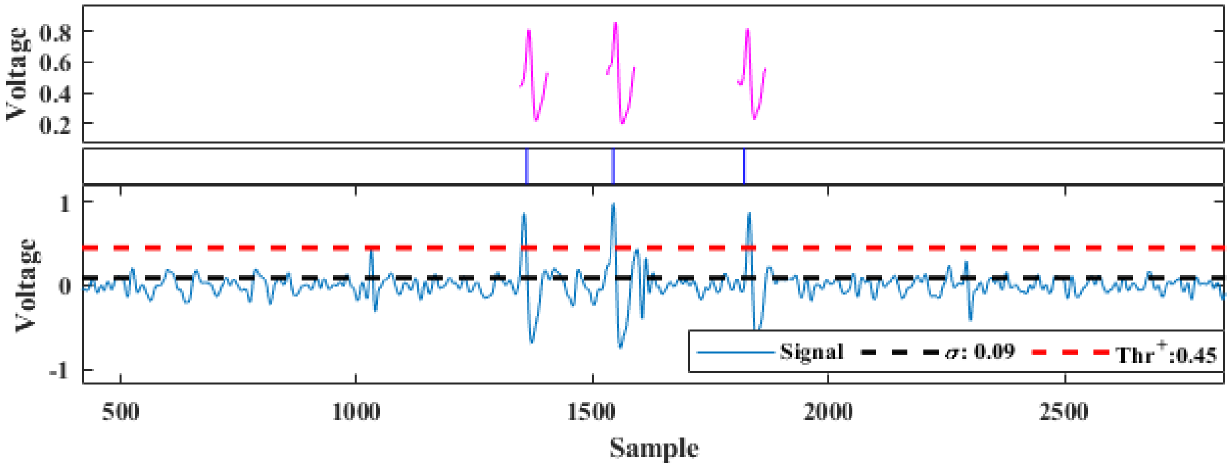
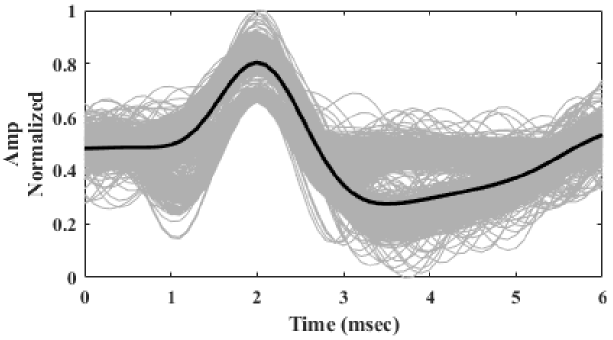
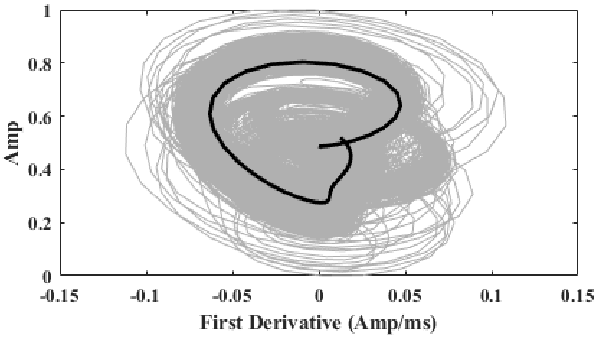


Fig. 1 shows spike waveforms (pink), raster plot (blue), and filtered signal (blue) with threshold (red, Thr = 0.45) and std dev (black, σ = 0.09) over 1.4 × 10^4 samples. Fig. 2 zooms in on Fig. 1's filtered signal from sample 500 to 2500.





The figures depict spike waveforms and phase space visualizations from the Spike Extraction Software. The first figure illustrates normalized amplitude versus time, showing multiple spike waveforms as gray traces and a typical spike waveform as a black trace, highlighting the temporal structure of detected spikes with amplitudes from 0 to 1 over a 0 to 6 millisecond period. The phase space plot, displaying normalized amplitude against its first derivative, illustrates the dynamic behavior of a spike as a closed-loop trajectory. This visualization, which can be saved, provides insights into the spike's rate of change and supports feature extraction for clustering, contributing to the analysis of single-channel extracellular recordings.

🔗 Explore the repository here: [Neurophysiological Signal Processing and Analysis]

Stay updated by following @rezasaadatyar