DATA-DRIVEN MULTI-CHANNEL FILTER DESIGN WITH PEAK-INTERFERENCE SUPPRESSION FOR THRESHOLD-BASED SPIKE SORTING IN HIGH-DENSITY NEURAL PROBES

Jasper Wouters [1] *Fabian Kloosterman* [2, 3, 4] *Alexander Bertrand* [1]

[1] KU Leuven, Electrical Engineering Dept. (ESAT),
Stadius Center for Dynamical Systems, Signal Processing, and Data Analytics, Belgium
[2] Neuro-Electronics Research Flanders (NERF), Leuven, Belgium
[3] KU Leuven, Brain & Cognition Research Unit, Belgium [4] VIB, Leuven, Belgium

ABSTRACT

Spike sorting is the process of assigning each detected neuronal spike in an extracellular recording to its putative source neuron. A linear filter design is proposed where the filter output allows for threshold-based spike sorting of high-density neural probe data. The proposed filter design is based on optimizing the signal-to-peak-interference ratio for each detectable neuron in a data-driven way. Threshold-based spike sorting using linear filters is particularly interesting for real-time spike sorting because of the low computational complexity and predictable delay of those filters, enabling closed-loop neuroscience with unit-activity controlled brain stimulation. We validate our method on both paired and hybrid in-vivo recorded high-density data.

***Index Terms*—** Real-time spike sorting, high-density probes, matched filtering, interference suppression

1. INTRODUCTION

High-density (HD) neural probes, containing hundreds of closely spaced electrodes [1], enable simultaneous recording of electrical activity from thousands of neurons located near the probe. Neuronal electrical activity consists, among other contributions, of all-or-none action potentials, also called spikes, which neurons generate to communicate with each other. Each electrode on the probe typically observes spikes from multiple neurons, which may overlap in time, and every neuron is typically also observed on multiple electrodes. Therefore, in order to study the neural activity on a per-neuron basis, the single-unit activity, i.e., the spike times of individual neurons, need to be retrieved from probe signals. This source separation process is referred to as *spike sorting* [2] [3].

Spike sorting is based on the assumption that every neuron generates a characteristic spatio-temporal spike waveform on the probe, and that this waveform is stable over different observations of spikes from the same neuron. Numerous spike sorting algorithms have been developed exploiting this assumption [4] [5] [6] [7] and many of these implement a similar processing pipeline: first spikes are detected in the recorded data, then features are extracted from those spikes, next a clustering algorithm is applied on these features, and finally previously unseen spikes can be classified using the clustering information.

A recurring problem using such a processing pipeline is that overlapping spikes are typically incorrectly classified, or even worse, consistent overlap might result in the occurrence of spurious clusters in the clustering phase, which might mistakenly be interpreted as an additional neuronal source. To solve this problem, the typical spike sorting pipeline can be augmented with a matched filtering stage, also referred to as template matching [8] [9] [10], where neuron-specific templates are estimated based on the clustering information, after which classification is done using linear template matching filters where usually iterative deflation techniques are used to resolve overlapping spikes [11] [12].

Classification based on linear filtering is also interesting for real-time spike sorting. [13] Real-time spike sorting promises to revolutionise neuroscience by enabling closed-loop experiments [14], where stimulation can be controlled using single-unit activity information. In such a context low and deterministic computational complexity is required. Hence, it is interesting to classify spikes according to their putative neurons in a single shot, i.e., non-iteratively using a thresholding operation on the per-neuron linear filter outputs.

A novel data-driven filter design is proposed which results in filters suitable for threshold-based spike sorting. The filters will maximize the *signal-to-peak-interference* ratio (SPIR), aiming to minimize the influence of the dominant interfering

This work was carried out at the ESAT Laboratory of KU Leuven, in the frame of KU Leuven Special Research Fund projects C14/16/057 and CoE PFV/10/002 (OPTEC), and the Research Foundation Flanders (FWO) project FWO G0D7516N (Distributed signal processing algorithms for spike sorting in nextgeneration high-density neuroprobes). The scientific responsibility is assumed by its authors.

spikes, as opposed to conventional matched filters focussing on optimizing the signal-to-noise ratio (SNR).

Matched filters taking into account interfering spike templates from other neurons as well as noise, have already been proposed [15] [16]. In practice however these methods will often lead to filters which still optimise SNR, because no strategy for weighting the different interfering sources is proposed, and the degrees of freedom of the filter are "wasted" on minimizing background noise.

In Section 2 the filter design procedure is introduced. Section 3 presents experiments applying the proposed filter design on both simulated and in-vivo acquired data. Section 4 will conclude this paper.

2. FILTER DESIGN METHODOLOGY

In this section, we explain our proposed data-driven multi-channel filter design method for threshold-based spike sorting. The filter design typically happens offline, where the filter is trained on some initial training data, after which it can be used for real-time spike sorting on new incoming data.

2.1. SPIR maximization

Consider $\mathbf{x}[t] \in \mathbb{R}^N$ to be an observation of the high-pass filtered extracellular potentials measured with an N-channel probe at discrete time t. The cutoff frequency f_c of the high-pass filters is set to $f_c = 300Hz$ such that spiking neurons become dominant signal sources in the filtered signal. Due to high-pass filtering, we can assume that $\mathbf{x}[t]$ has zero-mean.

Now let $\bar{\mathbf{x}}_k[t] = [\, x_k[t] \;\; x_k[t-1] \;\; \ldots \;\; x_k[t-L+1] \,]^T \in \mathbb{R}^L$ denote the observation of an L-taps delay line for the k^{th} channel of the probe at time t.
Stacking such delay lines for each channel k results in $\bar{\mathbf{x}}[t] = \left[\bar{\mathbf{x}}_1[t]^T \;\; \bar{\mathbf{x}}_2[t]^T \;\; \ldots \;\; \bar{\mathbf{x}}_N[t]^T \right]^T \in \mathbb{R}^{NL}$.

As spikes generated by the same neuron typically exhibit the same spatio-temporal 'signature' waveform across the probe, a spike template for a specific target neuron can be estimated by averaging M example spikes, i.e.,

$$\boldsymbol{\tau} = \frac{1}{M} \sum_{i=1}^{M} \bar{\mathbf{x}}[t_i]. \tag{1}$$

where $\{t_1, \ldots, t_M\}$ denote the example spike times, ensuring that all spikes are temporally aligned. In practice, example spike times are obtained from an initial spike clustering stage [11], where each resulting cluster contains example spikes corresponding to a specific target neuron.

For threshold-based spike sorting, our aim is to design an optimal linear multi-channel filter for each of these target neurons/clusters, which maximizes the margin between the filter's peak response to spikes from the target neuron and the filter's peak response due to other interfering sources. An interfering source is considered harmful if it has a high probability to generate a peak signal at the filter output that is larger or similar to the typical peak-response amplitude of the target neuron spike, thereby incorrectly exceeding the threshold. This may happen if the signal has a large amplitude in the channels near the target neuron, and/or if it has a similar spatio-temporal structure across the probe as the target neuron template $\boldsymbol{\tau}$. In the sequel, we will refer to such potentially harmful interfering sources as 'peak-interferers', and we will explain in Section 2.2 how to identify signal segments with such potentially harmful sources. Besides such peak-interference there is also less harmful background noise, such that $\mathbf{x}[t] = \mathbf{s}[t] + \mathbf{i}[t] + \mathbf{b}[t]$, with $\mathbf{s}[t]$ the signal generated by the target neuron which is summarized by $\boldsymbol{\tau}$, $\mathbf{i}[t]$ the signal generated by peak-interferers and $\mathbf{b}[t]$ background noise.

The optimal filter coefficients $\mathbf{f}$ that maximize the SPIR are then given by:

$$\mathbf{f} = \underset{\mathbf{w}}{\operatorname{argmax}} \frac{\left|\mathbf{w}^T \boldsymbol{\tau}\right|^2}{E\left\{\left|\mathbf{w}^T \bar{\mathbf{i}}[t]\right|^2\right\}} \quad \text{s.t. } \|\mathbf{w}\| = 1, \tag{2}$$

with $\mathbf{w} \in \mathbb{R}^{NL}$. Note that $\mathbf{f}$ is only defined up to a scaling, which is why we added the unit-norm constraint, which is an arbitrary choice. The closed-form solution of this problem can be straightforwardly found (e.g., using Lagrange multipliers), and is given by

$$\mathbf{f} = \frac{\mathbf{R}_{\bar{\mathbf{i}}\bar{\mathbf{i}}}^{-1} \boldsymbol{\tau}}{\|\mathbf{R}_{\bar{\mathbf{i}}\bar{\mathbf{i}}}^{-1} \boldsymbol{\tau}\|}, \tag{3}$$

with $\mathbf{R}_{\bar{\mathbf{i}}\bar{\mathbf{i}}} = E\left\{\bar{\mathbf{i}}[t] \;\; \bar{\mathbf{i}}[t]^T\right\}$ the peak-interference covariance matrix which can be estimated over signal segments exhibiting peak-interference, which will be further explained in Section 2.2. Note that $\mathbf{R}_{\bar{\mathbf{i}}\bar{\mathbf{i}}}$ might be ill-conditioned, in practice this problem is often solved by considering only a subset of electrodes closest to the target neuron or by applying a diagonal loading to the interference covariance matrix.

Spikes for the target neuron can then be detected by thresholding the filter output $y[t] = \mathbf{f}^T \bar{\mathbf{x}}[t]$. Without loss of generality, it is assumed that the maximal absolute filter response to a target spike is reached when the output has a positive value. Given a suitable threshold Thr, spikes of the target neuron are detected when $y[t] > Thr$.

Designing and applying such a filter for *every* neuron for which its spike waveforms are detectable in the recording, and thresholding the individual filter outputs, leads to a filter bank with single-unit activity at each filter's output.

2.2. Estimating peak-interference covariance $\mathbf{R}_{\bar{\mathbf{i}}\bar{\mathbf{i}}}$

The proposed filter design requires the interference covariance matrix to be known. A covariance estimate can be obtained as follows:

$$\hat{\mathbf{R}}_{\bar{\mathbf{i}}\bar{\mathbf{i}}} = \frac{1}{m} \sum_{j=1}^{m} \bar{\mathbf{i}}[t_j] \;\; \bar{\mathbf{i}}[t_j]^T, \tag{4}$$

but note that signals from interfering sources $\mathbf{i}[t]$ are not readily available.

Next an interference sensing scheme is presented, which is also visualized in Figure 1, for finding segments of $\mathbf{x}[t]$ where $\mathbf{i}[t] \neq \mathbf{0}$:

1. **Required prior knowledge**
   The interference sensing scheme makes use of a matched filter based on the spike template only, and as such requires the spike template $\boldsymbol{\tau}$ for the target neuron to be available (e.g., from prior clustering). The spike times $\{t_1, \ldots, t_M\}$ used for estimating the spike template are also assumed to be known.

2. **Template matching**
   Calculate the template matching output:
   $$z[t] = (\boldsymbol{\tau}^n)^T \bar{\mathbf{x}}[t]. \quad (5)$$
   This template matching output can be viewed as an initial 'naive' matched filtering operation, which is optimal if $\mathbf{i}[t] = \mathbf{0}$ and $\mathbf{b}$ is white across space and time. It will have a high output variance at times $\{t_1, \ldots, t_M\}$, but also at times when the neuron is spiking which are not included in the template estimation set, and at times when peak-interfering sources are active (either due to their high amplitude or due to having a similar spatio-temporal structure as the template $\boldsymbol{\tau}$).

3. **Target safe zone**
   To prevent leakage of the target neuron's spike covariance in the interference covariance matrix, a target safe zone is defined. The target safe zone is the interval given below:
   $$\left((1-\alpha)\min_i(z[t_i]), (1+\alpha)\max_i(z[t_i])\right), \quad (6)$$
   with $t_i \in \{t_1, \ldots, t_M\}$ and $\alpha \in \mathbb{R}^+$ a parameter to widen the target safe zone, e.g., $\alpha = 0.1$ yields good results in practice.

4. **Noise floor**
   Next the noise floor b of $z[t]$ is estimated. To this end a median operator is used, such that $b = \text{median}_t(|z[t]|)$

5. **Interference threshold**
   Finally the interference threshold Thr_{int} is chosen somewhere between the lower bound of the target safe zone and the noise floor:
   $$Thr_{int} = \beta(1-\alpha)\min_i(z[t_i]) + (1-\beta)b, \quad (7)$$
   with $\beta \in [0,1]$ a tuning parameter, e.g., $\beta = 0.5$.

6. **Interference segments**
   First using the interference threshold, candidate interference segments are identified when $z[t_j] > Thr_{int}$, over all t_j.

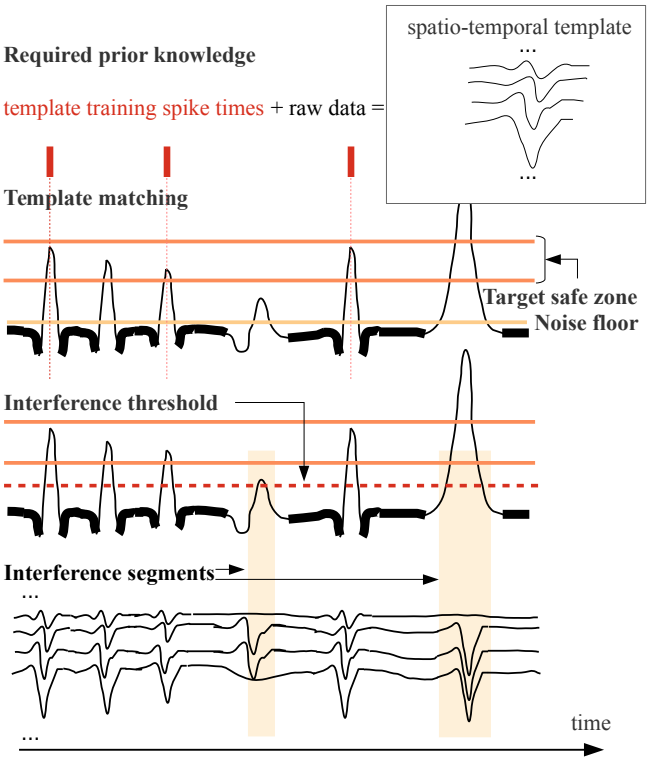

Fig. 1. Graphical representation of the interference sensing scheme.

   Next for all candidate segments the local neighbourhood $(t_j - \epsilon, t_j + \epsilon)$ is scanned for a local maximum of $z[t]$. ϵ is chosen to be the number of samples corresponding to a duration of $1ms$, which is the typical duration of a spike. If a local maximum is found and it is within the target safe zone, the candidate segment at t_j is rejected.

Replacing $\bar{\mathbf{i}}$ by $\bar{\mathbf{x}}$ in (4) is then used to estimate the interference covariance over all t_j retained from the interference sensing scheme.

It is possible that no candidate segments are retained, indicating that there are no dominant interfering sources. The interference covariance matrix can then be estimated over all segments for which the template matching output amplitude is not within the target safe zone, effectively optimizing the SNR with respect to the background noise $\mathbf{b}[t]$.

2.3. Choosing spike sorting threshold

The prior knowledge used to estimate the spike template $\boldsymbol{\tau}$ can also be used to determine a spike sorting threshold Thr. Given a matched filter as in (3), the threshold Thr can be determined by applying step 4 and 5 of the interference sensing scheme to $y[t]$ instead of $z[t]$. The spike sorting threshold Thr will then be given by (7).

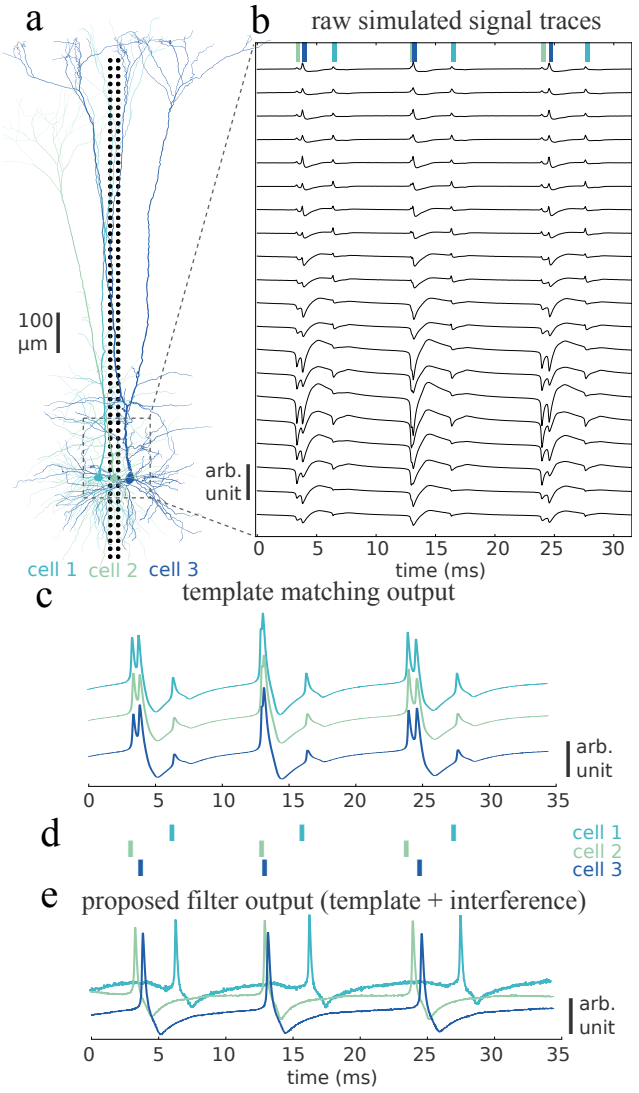

Fig. 2. **a**. Visualisation of the neurons' morphology and their relative position/orientation with respect to the probe. Each black dot marks the location of an electrode on the probe. **b**. Sample of the simulated extracellular potentials measured on a subset of 20 electrodes. The colored blocks on top code for the identity of the spiking neuron. **c**. Output from three matched filters (color coded by target neuron), each one trained using only the spike template of the target neuron. **d**. Ground truth spike trains for each neuron. **e**. Color coded output from three linear filters as proposed in this paper.

3. EXPERIMENTS

3.1. Illustrative example

The extracellular potentials impinging on a 128-channel HD probe originating from three spiking morphologically detailed neocortical layer 5b pyramidal cell models [17] are simulated [18] as shown in figure 2**a**-**b**. Because the data is acquired through simulation, ground truth spike times are readily available (Figure 2**d**). Spike templates are then calculated for each neuron using (1).

Given a spike template for each neuron, conventional template matching filters [10] are calculated similar to (5). The result from applying those filters directly to the data, is shown in Figure 2**c**. It can be observed that each filter output responds to all spikes from all neurons. As such, conventional template matching is not optimal for threshold-based spike sorting.

Now filters are trained for each neuron using the proposed filter design given by (3). The output for each filter is shown in Figure 2**e**. It can be seen from this figure panel that each filter output only responds to spikes generated by the filter's corresponding neuron. These filters allow for a threshold-based spike sorting of the simulated data.

3.2. Validation in-vivo recorded data

Having demonstrated the potential of the proposed filter design for threshold-based spike sorting, this filter design is applied to in-vivo recorded data for which ground truth spike times are available. Two types of ground truth data are used: paired recordings [19] and hybrid recordings [12]. The datasets used for this validation are available online [20] [21].

A total of 46 neurons for which ground truth spike times are available, were used for validating the use of the proposed filter for threshold-based spike sorting. The following spike sorting performance measurements are averages over all ground truth neurons, including data from both paired and hybrid data. The average sensitivity of the spike sorting, defined as $\frac{\text{true positives}}{\text{true positives + false negatives}}$, equals 88.3%. The average precision, defined as $\frac{\text{true positives}}{\text{true positives + false positives}}$, is equal to 94.7%. In comparison, using conventional template matching leads to a threshold-based spike sorting sensitivity of 81.0% and a precision of 89.1%.

The spike sorting thresholds Thr were tuned towards high precision rather than high sensitivity. Such a precision-sensitivity trade-off can be made by varying β in (7) in the context of Section 2.3.

4. CONCLUSION

A linear filtering threshold-based spike sorting framework applicable to online spike sorting was introduced. For such a framework to generate single-unit spike trains by a simple thresholding operation on the outputs of the filter bank, filters have to be designed that generate a high output variance only when the corresponding neurons are active. A filter design method was proposed that satisfies this requirement by optimizing the signal-to-peak-interference ratio. The filter design was tested on both simulated and real HD extracellular data and was shown to considerably improve the threshold-based spike sorting performance.

5. REFERENCES

[1] Carolina Mora Lopez et al., "22.7 a 966-electrode neural probe with 384 configurable channels in 0.13 μm soi cmos," in *Solid-State Circuits Conference (ISSCC), 2016 IEEE International*. IEEE, 2016, pp. 392–393.

[2] Michael S Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials," *Network: Computation in Neural Systems*, vol. 9, no. 4, pp. R53–R78, 1998.

[3] Sarah Gibson, Jack W Judy, and Dejan Marković, "Spike sorting: The first step in decoding the brain," *IEEE Signal processing magazine*, vol. 29, no. 1, pp. 124–143, 2012.

[4] Shy Shoham, Matthew R Fellows, and Richard A Normann, "Robust, automatic spike sorting using mixtures of multivariate t-distributions," *Journal of neuroscience methods*, vol. 127, no. 2, pp. 111–122, 2003.

[5] R Quian Quiroga, Zoltan Nadasdy, and Yoram Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering," *Neural computation*, vol. 16, no. 8, pp. 1661–1687, 2004.

[6] Shabnam N Kadir, Dan FM Goodman, and Kenneth D Harris, "High-dimensional cluster analysis with the masked em algorithm," *Neural computation*, 2014.

[7] Cyrille Rossant et al., "Spike sorting for large, dense electrode arrays," *Nature neuroscience*, vol. 19, no. 4, pp. 634–641, 2016.

[8] Olivier Marre et al., "Mapping a complete neural population in the retina," *Journal of Neuroscience*, vol. 32, no. 43, pp. 14859–14873, 2012.

[9] Chaitanya Ekanadham, Daniel Tranchina, and Eero P Simoncelli, "A unified framework and method for automatic neural spike identification," *Journal of neuroscience methods*, vol. 222, pp. 47–55, 2014.

[10] Felix Franke, Rodrigo Quian Quiroga, Andreas Hierlemann, and Klaus Obermayer, "Bayes optimal template matching for spike sorting–combining fisher discriminant analysis with optimal filtering," *Journal of computational neuroscience*, vol. 38, no. 3, pp. 439–459, 2015.

[11] Pierre Yger et al., "Fast and accurate spike sorting in vitro and in vivo for up to thousands of electrodes," *bioRxiv*, aug 2016.

[12] Marius Pachitariu et al., "Kilosort: realtime spikesorting for extracellular electrophysiology with hundreds of channels," *BioRxiv*, p. 061481, 2016.

[13] Felix Franke et al., "High-density microelectrode array recordings and real-time spike sorting for closed-loop experiments: an emerging technology to study neural plasticity," *Frontiers in neural circuits*, vol. 6, 2012.

[14] Davide Ciliberti and Fabian Kloosterman, "Falcon: a highly flexible open-source software for closed-loop neuroscience," *Journal of Neural Engineering*, 2017.

[15] William M Roberts and Daniel K Hartline, "Separation of multi-unit nerve impulse trains by a multi-channel linear filter algorithm," *Brain research*, vol. 94, no. 1, pp. 141–149, 1975.

[16] Roland Vollgraf and Klaus Obermayer, "Improved optimal linear filters for the discrimination of multichannel waveform templates for spike-sorting applications," *IEEE Signal Processing Letters*, vol. 13, no. 3, pp. 121–124, 2006.

[17] Etay Hay, Sean Hill, Felix Schürmann, Henry Markram, and Idan Segev, "Models of neocortical layer 5b pyramidal cells capturing a wide range of dendritic and perisomatic active properties," *PLoS computational biology*, vol. 7, no. 7, pp. e1002107, 2011.

[18] Henrik Lindén et al., "Lfpy: a tool for biophysical simulation of extracellular potentials generated by detailed model neurons," *Frontiers in neuroinformatics*, vol. 7, pp. 41, 2014.

[19] Joana P Neto et al., "Validating silicon polytrodes with paired juxtacellular recordings: method and dataset," *Journal of neurophysiology*, vol. 116, no. 2, pp. 892–903, 2016.

[20] Kampff Lab, "Ground-Truth data from silicon polytrodes," http://www.kampff-lab.org/validating-electrodes/, [Online; accessed 16-May-2017].

[21] Nick Steinmetz, "Index of /data/sortingComparison/datasets," http://phy.cortexlab.net/data/sortingComparison/datasets/, [Online; accessed 10-May-2017].