A novel unsupervised spike sorting implementation with variable number of features

Chaure F.J.[1,2,3,4], Quian Quiroga R.[2], Kochen S.S.[3], Rey H.G.[2]
[1]Instituto de Ingeniería Biomédica, UBA, C.A.B.A – Argentina, fjchaure@gmail.com
[2]Centre for Systems Neuroscience, University of Leicester, Leicester – UK
[3]Estudios de Neurociencias y Sistemas Complejos, CONICET - Hospital El Cruce - UNAJ, Florencio Varela – Argentina
[4]Instituto de Biología Celular y Neurociencias "Prof. E. De Robertis", Facultad de Medicina, UBA, C.A.B.A – Argentina

Abstract— **We propose a new fully automatic spike sorting algorithm that is able to match, or even improve, the performance of semiautomatic solutions with supervised intervention from expert users. We achieved this by incorporating: 1) a set of heuristic criteria inspired by the expert actions following the solution from semiautomatic algorithms, and 2) an improved feature selection method that increases the number of units that can be isolated from a single electrode recording. We evaluated the performance of the proposed method with real and simulated data.**

Index Terms— neurophysiology; signal processing, spike sorting

I. INTRODUCTION

Extracellular recordings of single neuron activity are done by placing electrodes in brain tissue. The electrical potential changes measured at the electrode tip reflect the activity of different neurons. By bandpass filtering the recorded signal, we have access to the spiking activity of a few neurons close enough to the electrode plus background activity elicited by neurons further away from the tip (black trace in the top panel of Fig. 1). In principle, the spikes fired by a neuron recorded by a given electrode have a distinct shape. This is mainly determined by its morphology, the distance and orientation relative to the recording site, and the properties of the extracellular medium [1]. Spike sorting algorithms detect these spikes (Fig. 1, top) and, using features extracted from the waveforms, group them into clusters that are associated with different neurons (Fig. 1, bottom) [2]. The importance of spike sorting is stressed by the fact that close-by neurons—whose firing is picked up by the same electrode—can fire in response to different things, such as stimuli or places, and therefore, it might be crucial to know which spike corresponds to which neuron [3].

Simultaneous recordings of large populations of neurons are becoming an essential tool for understanding complex behaviors and network properties in the brain [4]. Tetrodes have been used for over 25 years and other types of polytrode configurations and silicon probes have been developed in the last 10 years [5]. Large number of channels are currently used to record from local circuits in behaving animals [6]. With a high channel count, the manual supervision of each single channel might turn into a very time consuming task. In addition, the subjectivity introduced by the human intervention creates an additional source of sorting errors [7]. In this context, the unsupervised classification of single units has become the bottleneck to fully reach the potential of extracellular recordings [2,8].

Many spike sorting algorithms have been developed in the past years [2]. Although some methods propose online solutions [9], most of them rely on the offline processing on the data. Some of these methods are based on Bayesian statistical frameworks, relying in some cases on a Gaussian model of the distribution of the spike shapes [10]. Past studies, showing the non-Gaussian variability of the spike shapes and the non-stationarity nature of the extracellular recordings, have motivated the development of nonparametric approaches. One such method is Wave_clus [11], a spike sorting algorithm that uses wavelets decomposition to extract features of the spike waveforms and superparamagnetic clustering (SPC) to cluster the spikes in the feature space (see Appendix A for details of the current Wave_clus implementation).

Still, the most widely used algorithms are semiautomatic in practice, requiring manual tuning of the automatic solution by the expert to achieve a good performance. Particularly, we analyzed the actions performed by the expert using Wave_clus. We introduce in this work a set of heuristic modifications to reproduce the expert actions in a fully automatic way.

Another issue involving spike sorting algorithms is related to the number of units that can be discriminated from a single electrode recording. By recording simultaneous intra- and extra-cellular recordings, it has been showed that single neurons can be observed up to 50 μm from the cell body [12]. Then, there should be about hundred neurons that we should be identifying with single channel recordings. However, the number of neurons reported with single recording channels is generally below ten [13]. Several reasons have been suggested to explain the discrepancy between the number of neurons we should be observing and the ones that are actually observed [1,3], and deficits in spike sorting algorithms is one of them.

We propose to use a variable number of spike features automatically determined, so that when more neurons can be seen in the recording, more relevant features can be extracted to help discriminating those neurons. To evaluate the performance of this modification, we used the set of simulations introduced in [13], where recordings with up to 20 neurons were simulated.

II. PROPOSED DEVELOPMENTS TO IMPROVE WAVE_CLUS

In extracellular recordings, the activity of neurons with different firing rates and spike shapes generates clusters that differ in their density and location within the feature space. This condition may create clusters of different sizes that appear at different temperatures in the SPC (see Fig. 1). Moreover, in some cases Wave_clus finds automatically a temperature higher than the optimal one, leading to overclustering. Therefore, manual supervision from an expert is necessary to achieve a good sorting solution. To fix this limitation, we propose a new fully automatic method that outperformed the performance of the expert (as seen in Fig. 5).

A. Peak Selection in the temperature map

As the temperature is increased, a different data partition is generated by the SPC algorithm, and the appearance of new clusters is seen in the temperature map as an increase in one of the lines representing the cluster size at each temperature (see Fig. 1, bottom left). The rational of this first new step is to automatically detect all these "peaks" in the temperature map that show an increase of at least *N_inc* spikes.

Specifically, the proposed method first selects every cluster $C_i^{T_n}$ where its size $\left|C_i^{T_n}\right|$ is increased by at least *N_inc* spikes as the temperature is increased from $T_{n\text{-}1}$ to T_n, i.e.,

$$\left|C_i^{T_n}\right| - \left|C_i^{T_{n\text{-}1}}\right| \geq N\_inc, \qquad \text{with } \left|C_i^{T\text{-}1}\right| = 0 \tag{1}$$

To consider other relevant members of a partition at temperature T_n, all clusters $C_j^{T_n}$ with $j>i$ are also selected.

B. Inclusion criterion

By selecting all the candidate peaks as in the previous step, we capture all the potential clusters associated to the activity of single units. However, we will also increase the chance of overclustering and potentially select two clusters at different temperatures that could have large overlaps in their elements (and even have one included in the other). To overcome these difficulties, we added a second criterion which quantifies the overlap between pairs of clusters and results in the potential exclusion of one of them.

The Jaccard coefficient [14] measures similarity between clusters $C_i^{T_n}$ and $C_j^{T_m}$, and is given by $J_{i,j}=\left|C_i^{T_n} \cap C_j^{T_m}\right| / \left|C_i^{T_n} \cup C_j^{T_m}\right|$. The problem with this quantification is that it will be small even if $C_j^{T_m}$ is a subset completely included in $C_i^{T_n}$. Instead, we propose a modified version as an overlapping coefficient

$$O_{i,j}^{T_n,T_m} = |C_i^{T_n} \cap C_j^{T_m}| / \min(|C_i^{T_n}|,|C_j^{T_m}|) \tag{2}$$

This coefficient ranges between 0, if the clusters are completely different (i.e., they do not share any spike), and 1, when they are the same or one of them is a subset of the other. If the value of $O_{i,j}^{T_n,T_m}$ is not lower than a constant k_O, only the cluster at the higher temperature will be kept. In our implementation, we chose $k_O = 0.9$. An example of this criterion is shown in Fig. 2A, where a cluster was split in two.

The additional computational cost of this criterion is negligible compared to the cost associated already with the SPC algorithm.

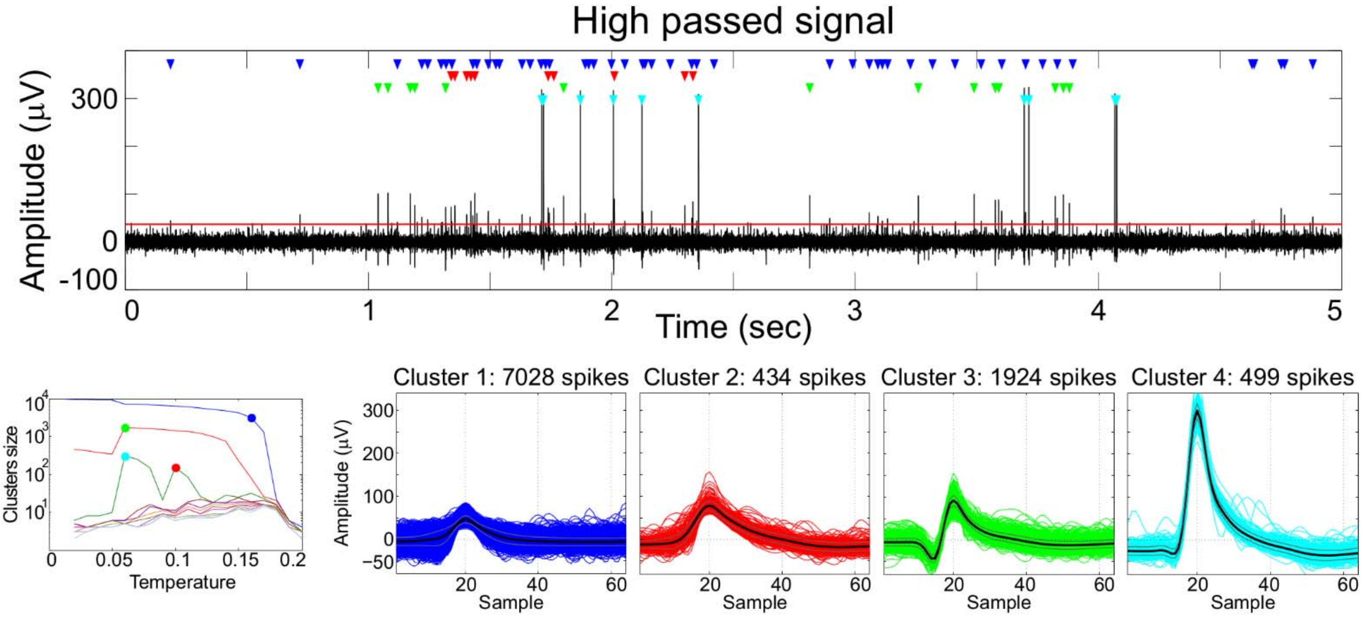

Fig. 1. Example of an extracellular recording from the human right entorhinal cortex. The black trace in the top panel shows the high frequency content of the signal (between 300 and 3000 Hz in this example). Neurons located more than ~140 microns away from the tip of the electrode contribute to the background noise, so their spikes cannot be detected. Closer neurons (between 50 and 140 microns away from the tip of the electrode) generate spikes larger than the background noise, but they cannot be isolated into different units, thus being associated to the multiunit activity (cluster 1). Finally, nearby neurons (less than ~50 microns) have even larger spikes, and sorting algorithms allow us to assign the recorded spikes to the different neurons that generated them (clusters 2–4). On the bottom left panel, the temperature map shows all the different partitions generated by the SPC, where filled circles denote the location of the clusters depicted on the right. In the top panel, the time of occurrence of each spike is marked with a triangle associated to the four isolated clusters. Adapted from [2].

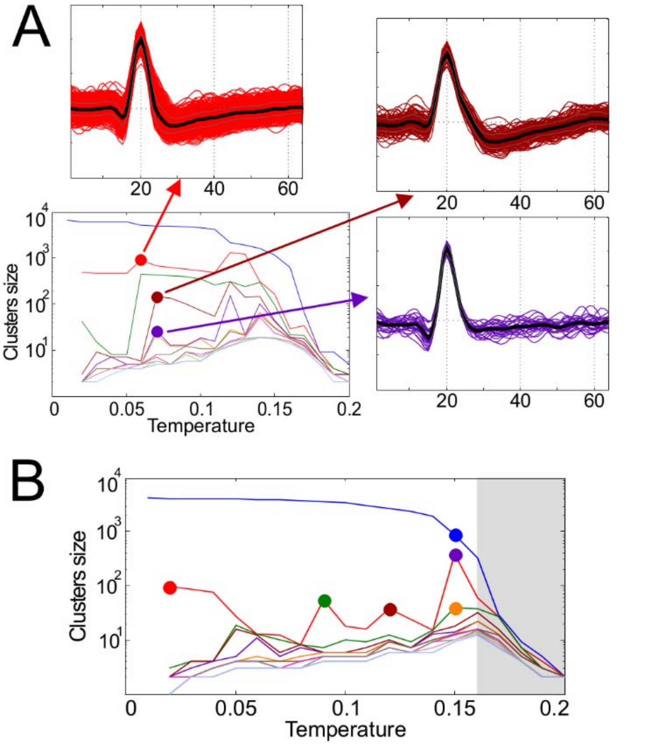

Fig. 2. New improvements introduced in the proposed method. **A.** Example of the inclusion criterion when a cluster was split in two at a higher temperature. The overlap coefficient between the marked clusters at T = 0.07 and the marked cluster at T = 0.06 was equal to one, so the clusters at T = 0.07 are retained, whereas the one at T = 0.06 is discarded. **B.** Example of the detection of the border of the superparamagnetic regime. T_B was identified at T = 0.16, so the partitions at $T_i \geq T_B$ were discarded (gray shaded area).

C. *Regime border detection*

As the temperature is increased and we move towards the end of the superparamagnetic regime, the chance of SPC generating overclustering is also increased. This is particularly worse when using small values on the parameter *N_inc*, but increasing it might cause the algorithm to miss small relevant clusters. The objective of the last criterion is to determine the temperature T_B where the superparamagnetic regime ends, discarding all the partitions appearing at $T_i \geq T_B$.

The end of the regime is normally associated with an abrupt decrease of the principal cluster as we transition from $C_1^{T_{B\text{-}1}}$ to $C_1^{T_B}$, with the appearance of several small clusters (overclustering). However, in some cases there is an abrupt decrease in the principal cluster in conjunction with the appearance of a large new and relevant cluster, typically being the second largest (C_2). For this reason, we first define the largest increment at a certain temperature as $LI_{T_i} = \max(|C_j^{T_i}|\text{-}|C_j^{T_{i\text{-}1}}|)$, for j>1. Then, we find the temperature T_B as,

$$T_B = \min\{T_i \mid Thr\_border > \left(|C_1^{T_i}|+LI_{T_i}\right)/|C_1^{T_{i\text{-}1}}|\} \qquad (3)$$

with *Thr_border* being a threshold parameter (we used a value of 0.4). This way, when the ratio is small, the principal cluster is largely split beyond what could have been related with a new cluster (associated to the increment *LI*). Fig. 2B shows an example of the identification of T_B, with the grey shaded area indicating all the partitions that are disregarded.

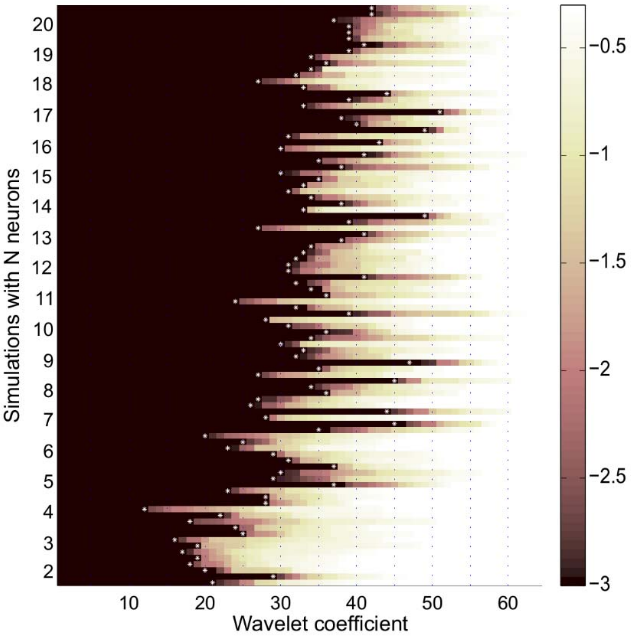

Fig. 3. p-values of the Lilliefors test on the distribution of each wavelet coefficient in the simulations of the simulated dataset. log10(p) is shown for each coefficient sorted in ascending order for each simulation. The white stars represent the number of coefficients that were selected on each simulation using the new criterion proposed. There is a clear correlation between the number of chosen coefficients and the number of simulated units (Spearman correlation, ρ = 0.64, p ~ 10^{-12}).

D. *Feature extraction (number of selected wavelet coefficients)*

Wave_clus selects the 10 most significant wavelet coefficients following a Lilliefors test. Fig. 3 shows the p-value of each coefficient (sorted by significance level) on each simulation in the simulated dataset. As more neurons are present in a recording, more features might be necessary to discriminate the differences in their corresponding waveforms. However, if we use all 64 coefficients, those capturing features of the noise might cause a huge drop in the clustering performance which is associated with the “curse of dimensionality” [15]. For our new implementation, we chose to include all the coefficients with a p-value equal to 10^{-3}, which is the smallest value tabulated in Matlab (Mathworks) for the Lilliefors test. As it can be seen in Fig. 3, this choice leads to a number of coefficients that correlates with the number of units.

III. Evaluation of the Proposed Method

A. *Performance with real data*

We first evaluated the performance of the proposed improvements with real data (see Appendix B). For each algorithm, we could compute a ROC (Receiver Operating Characteristic) curve (see Appendix C), as shown in Fig. 4.

In Wave_clus, as *N_inc* was increased, so did the true positive rate (TPR), with a reduction of the false positives (FP), until *N_inc* = 25; then the TPR stabilized at 75%–80%, without further improvement due to overclustering. As we included the

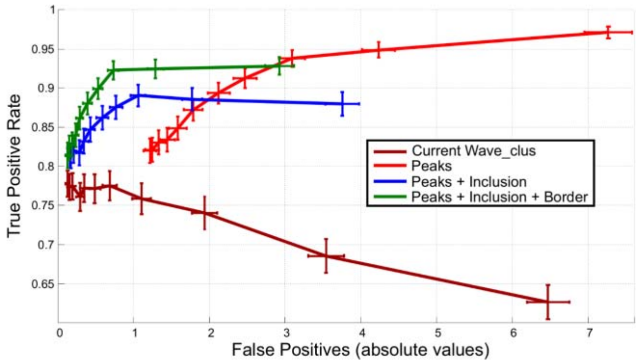

Fig. 4. Performance of the proposed improvements with a real dataset. For each algorithm, the parameter *N_inc* took values from 10 to 60 in steps of 5. In each case, the true positive rates and false positives were computed, allowing us to plot mean and standard error of the mean (SEM) across the 229 recordings analyzed. For the inclusion criterion, we used $k_O = 0.9$, and the border of the superparamagnetic regime was computed with *Thr_border* = 0.4.

proposed peak selection method, low values of *N_inc* were associated with high TPRs, but with many FPs. Increasing *N_inc* reduced the number of FPs (at the expense of reducing also the TPR), but remained large when compared with Wave_clus. However, the addition of the inclusion criterion led to the ROC being shifted to the left, but with a reduction in the TPR (although still with TPRs 10% higher than in Wave_clus) due to overclustering. Finally, the addition of the detection of the end of the superparamagnetic regime reduced the chance of overclustering, boosting the TPR an extra 5% while maintaining low FP values.

This dataset did not allow us to test the improvement associated with the selection of a variable number of wavelet coefficients, as the SPC solution used by the experts was done with a fixed number of 10 coefficients. Still, it allowed us to evaluate the performance of the other improvements introduced and assessed the contribution of each of them. Moreover, the results from Fig. 4 helped us to set the value *N_inc* = 20 as the choice for the new implementation, showing a good compromise between a high TPR and low number of FPs.

B. Performance with simulated dataset

In this case we used the dataset introduced in [13], where 5 simulations were done using N single units, with N = 2, 3,..., 20 (see Appendix D). Panels A and C in Fig. 5 show the performance of the algorithms in terms of hits/misses. As expected, the experts significantly improved the performance over the automatic solution from Wave_clus. As more neurons are simulated, there is also more room for differentiating the performance between algorithms. For this reason, panel C (and D) in Fig. 5 evaluate the performance (using paired sign tests)

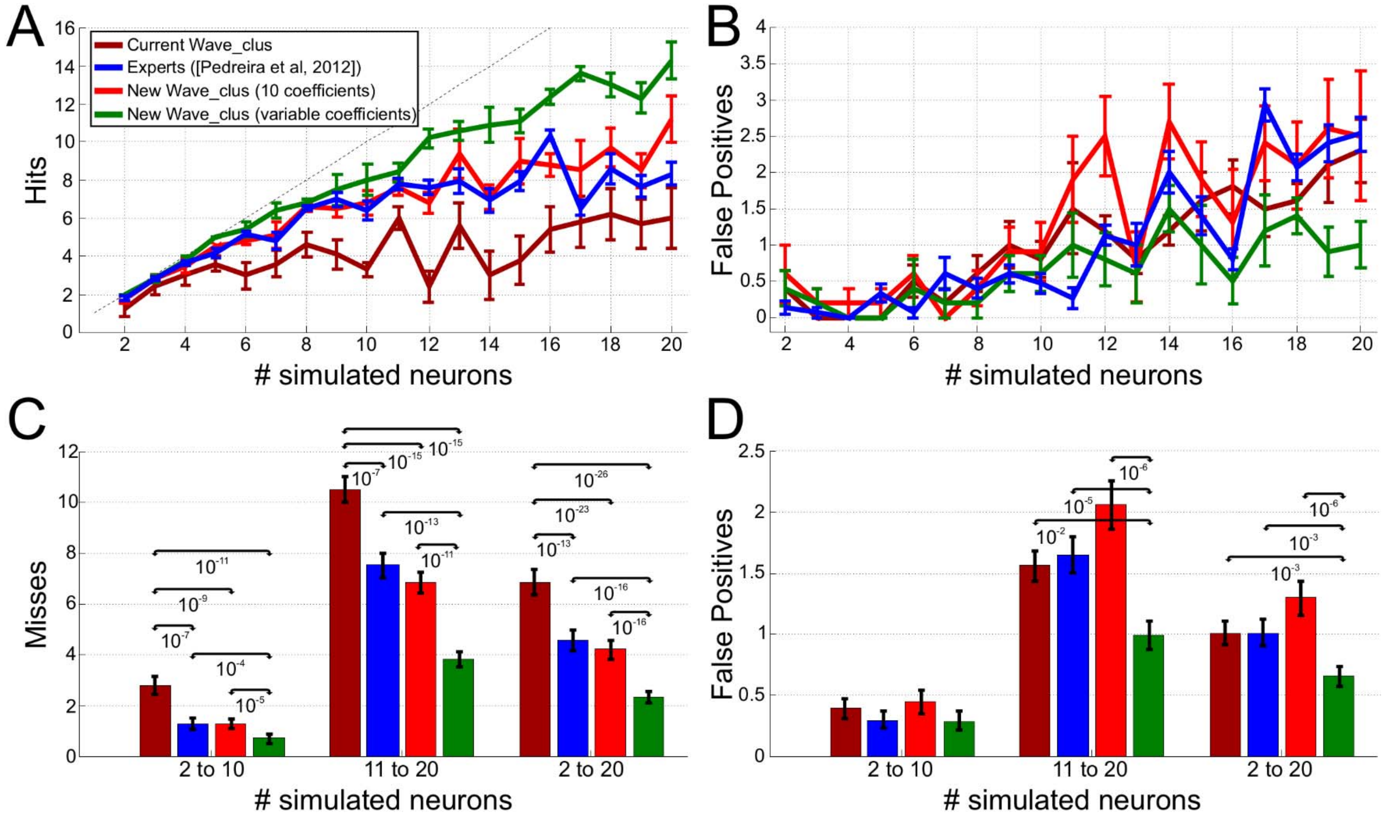

Fig. 5. Performance of each algorithm with the simulated dataset. **A.** Number of hits as a function of the number of simulated neurons for the different algorithm implementations. Mean and SEM across the 5 simulations with each number of simulated neurons is shown. **B.** Same as before but for the number of false positives. **C.** Mean and SEM of misses for each implementation. The analysis was done for the whole dataset ("2 to 20") and for the split datasets with a small ("2 to 10") and large ("11 to 20") number of neurons. Pair sign test were used to evaluate the statistical difference across implementations. All the implementations showed better performance than the unsupervised Wave_clus, whereas the proposed algorithm with a variable number of wavelet coefficients was significantly better than all other implementations. **D.** Same as C but with false positives. No differences were found in the subset with small number of neurons, but with more neurons (and the overall set), the proposed algorithm with a variable number of wavelet coefficients was significantly better than all other implementations.

across the whole set ("2 to 20"), but also separating the cases with a small ("2 to 10") and large ("11 to 20") number of neurons. When the proposed improvements are incorporated using 10 wavelet coefficients, the unsupervised performance was no different than the supervised one with Wave_clus. However, when a variable number of coefficient was used, as shown in Fig. 5, a further significant increase in the number of hits was achieved, outperforming all other methods in simulations with small and large number of neurons.

Still, these improvements can be at the expense of an increase in the number of false positives. As shown in panels B and D in Fig. 5, all algorithms behaved similarly in the cases with small number of neurons, keeping a small number of false positives. When more neurons were simulated, the number of false positives increased, but it was the new proposed implementation with variable wavelet coefficients the only method achieving a significantly better performance than all other methods. Therefore, the performance of the new proposed algorithm was better in terms of both hit and false positive rates.

IV. Discussion

The development of automatic and reliable spike sorting algorithms is becoming critical, given that within the next 10 years we will likely witness the number of recording sites going up to thousands [16]. The amount and complexity of the data to be produced by the next generation of probes is too large to be handled by researchers in a supervised way. In this context, the only viable option to fully take advantage of technological developments is to accompany them with the development of fully automatic spike sorting algorithms [17]. The proposed implementation introduced in this work goes in this line.

We presented a fully automatic method that significantly improved supervised performance based on the Wave_clus implementation. Most of the modifications introduced are inspired in actions performed by the expert. First, the algorithm looks for potential peaks in the temperature map at different temperatures. Then, two additional criteria remove clusters showing a large overlap with others (inclusion criterion) or appearing outside the superparamagnetic regime (regime border detection). Our method successfully overcame the problem of identifying clusters appearing at different temperatures, while avoiding overclustering. In addition, the variable number of wavelet coefficients provided the proposed method with the ability to use more features to discriminate more units, but keeping a low number of features when a small number of units is present or when the background noise is high. Moreover, as the selection is done in an unsupervised way, it shows improvement in cases in which the correct classification of spikes might turn very challenging for a human operator

Other algorithms have also been tested with the simulated dataset. In [13] the authors compared their results with Klustakwik [18], a spike sorting algorithm that extracts waveform features through a principal components analysis and use a Gaussian mixture model to perform the clustering of the data. The performance was similar to the one obtained with Wave_clus. More recently, the method Combinato has been introduced [19], which is based on the block-wise iterative use of SPC. Although their performance was better than the expert performance reported in [13], it is not as good as the one of the our new implementation. Particularly, the average number of hits they exhibited when 5, 10, 15, and 20 neurons were simulated were 4.5, 5.75, 9.25, and 10.5, respectively; in our algorithms, the results were 5, 8, 11, and 14, respectively.

The false positives found with the proposed method in the simulated dataset are related to overclustered units that have subtle differences between clusters and a size not large enough to generate a hit. The introduction of a merge procedure can potentially decrease the false positive rate while increasing the number of hits. This can be done based on the consequences of merging the clusters (e.g., evaluating cross-correlations, refractory period violations, etc.), and it can be done in an interactive way to reduce the time of manual intervention [10].

Appendices

A. Wave_clus implementation

Our new method is based on the current implementation of Wave_clus. As with for other spike sorting methods, this algorithm has four main steps: filtering, detection, feature extraction and clustering [2]. Zero-phase filtering is done by using a second-order bandpass elliptic filter in the range 300-3000 Hz. Spike detection is performed by setting a threshold as:

$$\mathrm{Thr} = 5\sigma_n, \text{ with } \sigma_n = \mathrm{median}\left\{\frac{|x|}{0.6745}\right\} \quad (4)$$

where x is the bandpass filtered signal [2]. An example of a signal filtered for detection and thresholding is shown in the top panel of Fig. 1. For each detected spike, 64 samples are saved for further analysis, aligned to their maximum at data point 20. To avoid spike misalignments due to low sampling, spike maxima location is refined by using cubic splines interpolated waveforms with 320 samples. After realignment, the waveforms are downsampled back to 64 points.

Feature extraction is done using a 4-scale multiresolution decomposition with a Haar wavelet, resulting in 64 wavelet coefficients for each detected spike. To assess the ability of each coefficient to separate different clusters a Lilliefors test is used, and retain only the 10 most significant ones. To minimize the effect of outliers in the test, values within ±3 standard deviations are the only ones considered for each coefficient.

Finally, a nonparametric clustering is performed in the feature space using SPC, an unsupervised approach in which the grouping of points into clusters mainly depends on the nearest-neighbor interactions [20]. The SPC generates a family of solutions as a function of a parameter called the temperature, which plays a major role in how the clusters are formed. In analogy with models in Statistical Mechanics, at low temperatures, all data points will be highly correlated and therefore will be grouped into a single cluster. On the other hand, at high temperatures, the large clusters will be divided into many groups of small size. At a certain temperature range between these two extremes, natural clusters will appear (i.e. the superparamagnetic regime) and only points corresponding to data from relatively high density regions will be grouped. For each temperature, a different data partition is generated with new clusters merging or breaking, which results in a "temperature map" (Fig. 1, bottom left).

In Wave_clus, clusters are automatically identified according to their size. The idea is that as one increases the temperature, new clusters will appear, as a subset separated from the bigger cluster. Wave_clus selects the highest temperature where at least one cluster increases its size a minimum number of spikes (parameter *N_inc*) [11]. At this temperature, all the clusters with a size bigger than *N_inc* are accepted.

In this work, SPC used a range of temperatures from 0 to 0.25 in increments of 0.01, where each temperature is labelled as T_i, with $i = \{0, \ldots, 25\}$. At each temperature, clusters are sorted in decreasing order with respect to their size; thus, at temperature T_i the largest cluster is denoted as $C_1^{T_i}$.

B. Real dataset: recordings in the human temporal lobe

Recordings in the human medial temporal lobe of 5 patients implanted with depth electrodes for epilepsy diagnosis were used, which were first reported in [21]. Each electrode probe had a total of nine microwires at its end, eight active recording channels and one reference. The differential signal from the microwires was amplified by a 64-channel Neuralynx system, filtered between 1 and 9,000 Hz and sampled at 28 kHz.

This dataset was sorted with Wave_clus and then manually supervised by an expert. Being real data, we do not have ground truth. However, the idea of the new method is to automatically replicate the steps manually done by the expert. Therefore, the expert's solution was used as "ground truth" to assess the performance of our new method. A total of 229 recordings were used (each approximately 30 minutes long), where at least two clusters were selected by the expert.

C. ROC analysis

Based on the temperature map of the real dataset, different selections of clusters were performed for different algorithms and parameter choices. The TPR was computed as the ratio between the number of clusters identified by the proposed algorithm and the number of clusters identified by the expert. In addition, the number of FP was assessed by counting the number of clusters identified by the proposed algorithm that were not associated to a cluster selected by the expert. This way, for each algorithm and *N_inc* value (ranging from 10 to 60 in steps of 5), we computed the TPR and FP on the real dataset, allowing us to construct ROC curves.

D. Simulated Dataset

A publicly available simulated dataset was used to evaluate the performance of the proposed method with ground truth. The set was presented in [13] , and it contains 5 simulations with *N* single units, where *N* = 2, 3,..., 20 (dataset available at http://bioweb.me/CPGJNM2012-dataset). In addition, three independent experts performed manual sorting on this set using Wave_clus. We averaged their performance to get a single "Expert" score per simulation.

The number of hits and false positives were quantified as in [13]. A selected cluster was considered as a hit when more than the 50% of its spikes corresponded to more than 50% of the spikes of the simulated cluster. In addition, a selected cluster was labelled as a false positive when it did not meet the hit criterion. Missed clusters were calculated as the number of simulated units minus the number of hits. The multiunit clusters were not considered as hits and they were considered as false positives when less than 50% of their spikes came from the multiunit.

References

[1] C. Gold, D. A. Henze, C. Koch, and G. Buzsáki, "On the origin of the extracellular action potential waveform: A modeling study," J. Neurophysiol., vol. 95, no. 5, pp. 3113–3128, May 2006.

[2] H. G. Rey, C. Pedreira, and R. Quian Quiroga, "Past, present and future of spike sorting techniques," Brain Res. Bull., vol. 119, pp. 106–117, Oct. 2015.

[3] H. G. Rey et al., "Single-cell recordings in the human medial temporal lobe," J. Anat., vol. 227, no. 4, pp. 394–408, Oct. 2015.

[4] R. Quian Quiroga and S. Panzeri, "Extracting information from neuronal populations: information theory and decoding approaches," Nat. Rev. Neurosci., vol. 10, no. 3, pp. 173–185, Mar. 2009.

[5] G. Buzsáki, "Large-scale recording of neuronal ensembles," Nat. Neurosci., vol. 7, no. 5, pp. 446–451, May 2004.

[6] A. Berényi et al., "Large-scale, high-density (up to 512 channels) recording of local circuits in behaving animals," J. Neurophysiol., vol. 111, no. 5, pp. 1132–1149, Mar. 2014.

[7] F. Wood, M. J. Black, C. Vargas-Irwin, M. Fellows, and J. P. Donoghue, "On the variability of manual spike sorting," IEEE Trans. Biomed. Eng., vol. 51, no. 6, pp. 912–918, Jun. 2004.

[8] K. D. Harris, R. Quian Quiroga, J. Freeman, and S. L. Smith, "Improving data quality in neuronal population recordings," Nat. Neurosci., vol. 19, no. 9, pp. 1165–1174, Aug. 2016.

[9] A. Oliynyk, C. Bonifazzi, F. Montani, and L. Fadiga, "Automatic online spike sorting with singular value decomposition and fuzzy C-mean clustering," *BMC Neurosci.*, vol. 13, p. 96, Aug. 2012.

[10] C. Rossant et al., "Spike sorting for large, dense electrode arrays," Nat. Neurosci., vol. 19, no. 4, pp. 634–641, Mar. 2016.

[11] R. Quian Quiroga, Z. Nadasdy, and Y. Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering," Neural Comput., vol. 16, no. 8, pp. 1661–1687, 2004.

[12] D. A. Henze et al., "Intracellular Features Predicted by Extracellular Recordings in the Hippocampus In Vivo," J. Neurophysiol., vol. 84, no. 1, pp. 390–400, Jul. 2000.

[13] C. Pedreira, J. Martinez, M. J. Ison, and R. Quian Quiroga, "How many neurons can we see with current spike sorting algorithms?," J. Neurosci. Methods, vol. 211, no. 1, pp. 58–65, Oct. 2012.

[14] P. Jaccard, "Étude comparative de la distribution florale dans une portion des Alpes et des Jura," Bull. Société Vaudoise Sci. Nat., vol. 37, pp. 547–579, 1901.

[15] C. M. Bishop, Pattern recognition and machine learning. New York: Springer, 2006.

[16] A. P. Alivisatos et al., "Nanotools for Neuroscience and Brain Activity Mapping," ACS Nano, vol. 7, no. 3, pp. 1850–1866, Mar. 2013.

[17] G. T. Einevoll, F. Franke, E. Hagen, C. Pouzat, and K. D. Harris, "Towards reliable spike-train recordings from thousands of neurons with multielectrodes," Curr. Opin. Neurobiol., vol. 22, no. 1, pp. 11–17, Feb. 2012.

[18] K. D. Harris, D. A. Henze, J. Csicsvari, H. Hirase, and G. Buzsáki, "Accuracy of Tetrode Spike Separation as Determined by Simultaneous Intracellular and Extracellular Measurements," J. Neurophysiol., vol. 84, no. 1, pp. 401–414, Jul. 2000.

[19] J. Niediek, J. Boström, C. E. Elger, and F. Mormann, "Reliable Analysis of Single-Unit Recordings from the Human Brain under Noisy Conditions: Tracking Neurons over Hours," PLOS ONE, vol. 11, no. 12, p. e0166598, 2016.

[20] M. Blatt, S. Wiseman, and E. Domany, "Data Clustering Using a Model Granular Magnet," Neural Comput., vol. 9, no. 8, pp. 1805–1842, Nov. 1997.

[21] R. Quian Quiroga, R. Mukamel, E. A. Isham, R. Malach, and I. Fried, "Human single-neuron responses at the threshold of conscious recognition," Proc. Natl. Acad. Sci., vol. 105, no. 9, pp. 3599–3604, Mar. 2008.