Journal of Neuroscience Methods 222 (2014) 47–55

Contents lists available at ScienceDirect

Journal of Neuroscience Methods

journal homepage: www.elsevier.com/locate/jneumeth

Computational neuroscience

A unified framework and method for automatic neural spike identification

Chaitanya Ekanadham[a,*], Daniel Tranchina[a,b], Eero P. Simoncelli[a,b,c]

[a] Courant Institute of Mathematical Sciences, New York University, United States
[b] Center for Neural Science, New York University, United States
[c] Howard Hughes Medical Institute, United States

ARTICLE INFO

Article history:
Received 19 December 2012
Received in revised form
30 September 2013
Accepted 2 October 2013

Keywords:
Spike sorting
Action potential
Neural spike identification
Clustering
Spike detection
Multi-electrode

ABSTRACT

Automatic identification of action potentials from one or more extracellular electrode recordings is generally achieved by clustering similar segments of the measured voltage trace, a method that fails (or requires substantial human intervention) for spikes whose waveforms overlap. We formulate the problem in terms of a simple probabilistic model, and develop a unified method to identify spike waveforms along with continuous-valued estimates of their arrival times, even in the presence of overlap. Specifically, we make use of a recent algorithm known as Continuous Basis Pursuit for solving linear inverse problems in which the component occurrences are sparse and are at arbitrary continuous-valued times. We demonstrate significant performance improvements over current state-of-the-art clustering methods for four simulated and two real data sets with ground truth, each of which has previously been used as a benchmark for spike sorting. In addition, performance of our method on each of these data sets surpasses that of the best possible clustering method (i.e., one that is specifically optimized to minimize errors on each data set). Finally, the algorithm is almost completely automated, with a computational cost that scales well for multi-electrode arrays.

The problem of detection, time-estimation, and cell classification of neural action potentials from extracellular electrode measurements is fundamental to experimental neuroscience. Electrode(s) are embedded in neural tissue, and a voltage trace is recorded as a function of time. When a neuron in the vicinity of the electrode fires an action potential, a stereotypical waveform is superimposed onto the recorded voltage (Lewicki, 1998; Sahani et al., 1997; Wehr et al., 1999). The shape of this waveform depends on the cell's morphology and position, as well as the filtering properties of the medium and the electrode(s). The "spike sorting" problem consists of detecting the occurrence of these individual waveforms and estimating their corresponding times of occurrence.

Despite the ubiquity and succinct formulation of the problem, there is no *de facto* standard for spike sorting. Traditionally, experimentalists manually position a single electrode and define threshold triggers to identify the spikes of individual cells (Rodieck, 1967). However, this becomes substantially more difficult when recording from several cells simultaneously, and is infeasible for multi-electrode arrays. Computer-assisted solutions have converged on a general methodology that we will refer to as "clustering," consisting of three steps (Lewicki, 1998), illustrated in Fig. 1: (1) detection of temporal segments of the voltage trace that are likely to contain spikes, (2) estimation of a set of features for each segment, and (3) classification of the segments according to these features. A variety of methods exist for solving each step (e.g., (1) thresholding based on absolute value (Obeid and Wolf, 2004), squared values (Rutishauser et al., 2006), Teager energy (Choi et al., 2006), or other nonlinear operators (Rebrik et al., 1999), (2) features such as peak-to-peak width/amplitude, projections onto principal components (Lewicki, 1998), or wavelet coefficients (Quiroga et al., 2004; Kwon and Oweiss, 2011), and (3) classification methods such as *K*-means (Lewicki, 1998), mixture models (Sahani, 1999; Shoham et al., 2003), or superparamagnetic methods (Quiroga et al., 2004)).

Although methods exist for solving each of the three steps in isolation, it is unclear how to relate the sequential application of these steps to the optimization of a single objective function, making it difficult to state the assumptions and operating conditions needed for success. Since each step does not take into account errors introduced in previous steps, errors tend to accumulate. In addition, many of these methods require human supervision (especially for the classification step), which is not only costly, but generally inaccurate (Harris et al., 2000) and highly variable (Wood et al., 2004). The lack of a standard automated methodology makes it difficult to compare results of scientific studies.

* Corresponding author. Tel.: +1 9146216070.
E-mail address: chaitue@gmail.com (C. Ekanadham).

0165-0270/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.jneumeth.2013.10.001

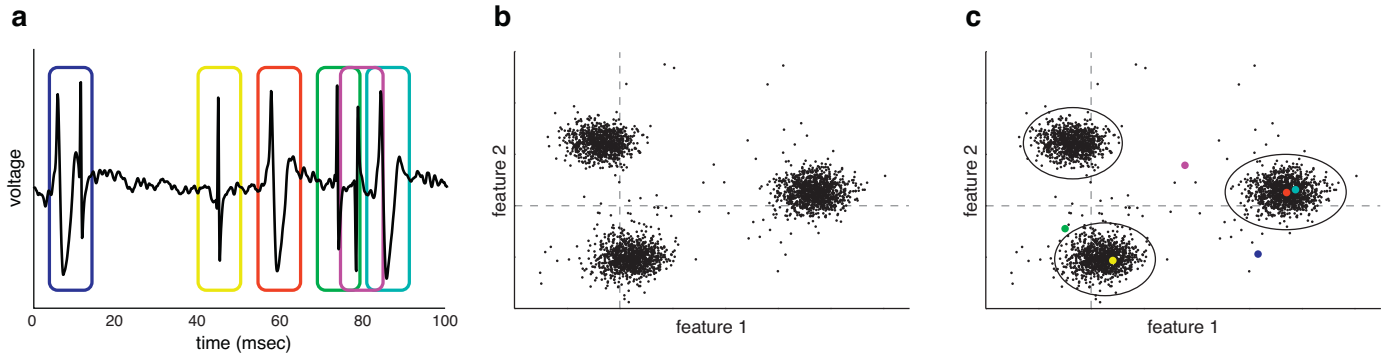

Fig. 1. Schematic of 3-step procedure common to most current spike sorting methods. (a) Thresholding/windowing. Voltage peaks are detected (by comparison to a threshold), and their occurrence times are estimated. Temporal segments of the voltage trace that lie within a fixed-duration window around each peak (colored rectangles) are gathered. (b) Feature estimation. Segments are projected into a low-dimensional feature space. Here, we plot the projection of each segment onto the first two principal components of the full set of segments. (c) Classification. Segments are grouped within the feature space, typically using an automatic clustering method such as *K*-means or estimation of a Gaussian mixture model. Colored points correspond to the windowed segments in (a). Note that several of these are mis-classified because they contain a superposition of more than one spike. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

Most importantly, the conventional three-step procedure mishandles overlapping spikes. If two or more cells fire near-synchronously, their respective waveforms will overlap in the voltage trace, creating a shape that differs from either spike in isolation (Lewicki, 1994; Sahani et al., 1997; Wehr et al., 1999; Pillow et al., 2013). If the waveform shapes partially cancel, the initial detection stage may miss the spikes altogether. Even if the segment is detected, its appearance will depend on the time delay between the two spikes (Pillow et al., 2013). If this is significantly different from that of either spike in isolation, it will be misidentified as a fictitious third cell or discarded as an outlier, as illustrated in Fig. 1. Even in the best case scenario, only one of the two spikes can be correctly identified by a clustering method, and the other discarded.

Failure to resolve overlapping spikes can have serious consequences: Basic measurements, such as mean firing rates and cross-correlations, can be heavily biased due to spike sorting artifacts (Bar-Gad et al., 2001; Pazienti and Grn, 2006; Pillow et al., 2013). Properly handling this bias is crucial when studying a neural population where there is a high level of synchronous activity or when the study itself focuses on the correlation of firing patterns (Mastronarde, 1989; Devries, 1999; Schnitzer and Meister, 2003; Shlens et al., 2008; Pillow et al., 2008). Such studies are more frequent with the advent of multi-electrode array recordings, which allow the simultaneous recording of large populations of neurons (Meister et al., 1994; Gerstein and Clark, 1964; Brown et al., 2004; Pillow et al., 2008; Shlens et al., 2009).

There have been several proposed methods to augment the clustering approach to account for overlapping spikes (Atiya, 1992; Lewicki, 1994; Segev et al., 2004; Zhang et al., 2004; Vargas-Irwin and Donoghue, 2007; Pillow et al., 2008; Chen et al., 2011; Prentice et al., 2011; Pillow et al., 2013). However, these methods generally rely on brute-force examination of all combinations of spike waveforms at all time separations (impractical for simultaneous recordings of many cells), or "greedy" algorithms that iteratively subtract the waveform of the best-fitting cell until the residual amplitude is within the range expected for noise. A notable exception is the family of ICA-based spike sorting methods (Takahashi et al., 2003; Takahashi and Sakurai, 2005; Franke et al., 2009), which bear some resemblance to our approach, but have not been developed or implemented in the context of a unified probabilistic model for the voltage measurements, and have not been extensively tested and compared to traditional clustering methods.

In this paper, we present a method for estimation of the most probable spike patterns given the observed voltage trace, which is assumed to be a noisy linear superposition of spike waveforms shifted to their respective spike times, corrupted by additive noise. We use a recently developed method in sparse signal decomposition, known as *Continuous Basis Pursuit* (Ekanadham et al., 2011a), as the basis for an accurate and efficient approximation of the solution. The resulting method provides a unified procedure for the estimation of continuous-valued spike times that operates correctly in the presence of overlapping spikes, and does not rely on any auxiliary heuristic pre-processing or post-processing such as alignment of spike segments or searching for spike combinations. An initial version of this work was presented in (Ekanadham et al., 2011b). A software implementation is available at http://www.cns.nyu.edu/~lcv/spikeSorting.html

1. Methods

1.1. Spike sorting using Continuous Basis Pursuit (CBP)

Our method is derived from a simple generative model for the observed voltage trace (Sahani, 1999; Pillow et al., 2008, 2013), as illustrated in Fig. 2. A spike from the nth neuron, occurring at time $\{\tau_{ni}\}$, is assumed to produce a temporally localized waveform $a_{ni}W_n(t-\tau_{ni})$, where $W_n(t)$ has unit norm, and a_{ni} represents the (root-mean-squared) spike amplitude. These time-shifted and scaled waveforms are then added together with noise to form the electrode voltage trace:

$$V(t) = \sum_{n=1}^{N}\sum_{i=1}^{C_n} a_{ni}W_n(t-\tau_{ni}) + \epsilon(t) \quad (1)$$

In the case of multi-electrode recordings, $V(t)$ and $W_n(t)$ are vector-valued with as many dimensions as electrodes, but for notational convenience, the derivation below is written for the scalar case.

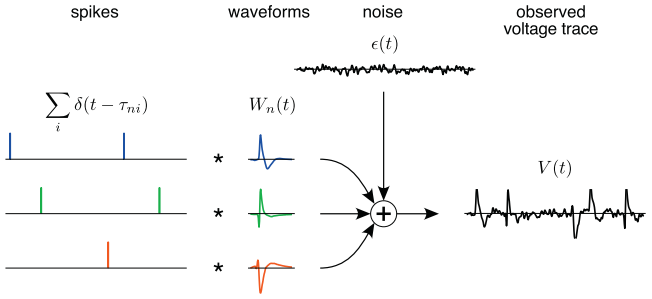

Fig. 2. Measurement model, illustrated for three neurons and a single electrode. Each cell generates a voltage trace containing time-shifted copies of its spike waveform, and the observed voltage trace is assumed to be a sum of these and noise.

The distribution of the noise, $\epsilon(t)$, is assumed to be log-concave, which leads to a tractable optimization algorithm, while allowing for Gaussian or more heavy-tailed distributions (e.g., Laplacian or power-law) that arise in many experimental settings (Sahani, 1999; Shoham et al., 2003; Fee et al., 1996). For the data sets analyzed in this article, we found that our method worked well with an assumed Gaussian distribution, so we restrict ourselves to that case from here on. Our approach can also handle correlated noise, but to simplify the derivation, we assume any such correlations have been removed through pre-processing step (see Section 1.3). Given these assumptions, the goal is to recover the parameters $\{W_n(t)\}_{n=1}^{N}, \{\tau_{ni}\}_{i=1}^{C_n}, \{a_{ni}\}_{i=1}^{C_n}$ given the voltage trace $V(t)$.

1.1.1. MAP estimation

We aim to compute the most probable solution of the waveforms, spike times, and spike amplitudes given the observed voltage trace $V(t)$, i.e. the *maximum a posteriori* (MAP) solution, given by:

$$\begin{aligned}&\operatorname{argmax}_{\{a_{ni}\},\{\tau_{ni}\},\{W_n(t)\}} P(\{a_{ni}\},\{\tau_{ni}\}|V(t);\{W_n(t)\})\\ &= \operatorname{argmax}_{\{a_{ni}\},\{\tau_{ni}\},\{W_n(t)\}} \log(P(V(t)|\{a_{ni}\},\{\tau_{ni}\};\{W_n(t)\}))\\ &+ \log(P(\{a_{ni}\},\{\tau_{ni}\}))\end{aligned} \tag{2}$$

Note that the waveforms are treated as deterministic model parameters, since we do not assume any prior probability distribution (although such *a priori* information could be incorporated if desired). Using the model of Eq. (1), and assuming the noise $\epsilon(t)$ is Gaussian with mean zero, Eq. (2) can be rewritten as:

$$\begin{aligned}&\operatorname{argmin}_{\{a_{ni}\},\{\tau_{ni}\},\{W_n(t)\}} \frac{1}{2}\|V(t)-\sum_{n,i} a_{ni}W_n(t-\tau_{ni})\|_2^2\\ &- \log P(\{a_{ni}\},\{\tau_{ni}\})\end{aligned} \tag{3}$$

This optimization problem partitions naturally into two subproblems: solving for the waveform shapes, $\{W_n(t)\}$, and solving for the spike amplitudes and times, $\{a_{ni}, \tau_{ni}\}$. Thus, we use a "coordinate descent" algorithm, alternating between solving for each of these two subsets of parameters while holding the other subset fixed.

1.1.2. Solving for spike times/amplitudes: CBP

Given the waveforms, solving directly for the spike times/amplitudes is intractable, because the spike times are embedded as continuous arguments within the nonlinear waveform functions. We use a recently developed sparse optimization method known as *Continuous Basis Pursuit* (CBP) (Ekanadham et al., 2011a), which is designed to handle superpositions of localized waveforms that occur at arbitrary times, coupled with a sparsity-inducing penalty on the amplitudes (Candes, 2008). This reduces the problem to solving a simpler objective function:

$$\operatorname{argmin}_{\vec{x}\in\mathcal{C}} \frac{1}{2}\|V(t)-\mathbf{W}\vec{x}\|_2^2 + \sum_{n,i} p\log(\epsilon + x_{ni1}). \tag{4}$$

Here, $\mathbf{W}$ is a set of basis functions that approximately span the space of continuously time-shifted versions of the waveforms $\{W_n(t-\tau)\}$, and $\vec{x}$ is a vector of coefficients, constrained by inequalities, that encode the spike times and amplitudes (Ekanadham et al., 2011a). Specifically, the existence, amplitude, and time of a spike of the nth waveform occurring within the ith time bin is encoded by a group of three real-valued coefficients $\{x_{nij}\}_{j=1,2,3}$. The first coefficient x_{ni1} is constrained to be positive, and encodes the spike amplitude (0 indicates no spike). The remaining two coefficients in each group encode the time-shift within the time bin. The convex constraint set, $\mathcal{C}$, is defined by a set of inequalities that ensure that the first (amplitude) coefficient is positive, and that the other two coefficients are bounded, such that they are only used when the first coefficient is nonzero, and their effect is to (approximately) time-shift the waveform within the time bin (see Ekanadham et al., 2011a). The second term in Eq. (4) is a sparsity-promoting penalty term which biases the solution toward zero-valued coefficients (since we expect only a small proportion of time bins to contain spikes). Probabilistically, this corresponds to an assumption of a power-law pseudo-prior on the amplitudes ($\mathrm{P(a)} \propto 1/(\epsilon+a)^p$). We found that performance was relatively insensitive to the choice of parameters p and ϵ, and for all results reported here, the values were set to 10 and 10^{-16}, respectively.

We solve Eq. (4) using iteratively reweighted L_1 minimization (Candes, 2008). Briefly, we employ a first-order Taylor expansion of the nonconvex log prior term about a previous estimate of $\vec{x}$:

$$\begin{aligned}p\log(\epsilon+x) \approx p\Bigg(&\log(\epsilon+x_{ni1}^{(k-1)}) + \frac{d}{dx}\log(\epsilon+x)|_{x=x_{ni1}^{(k-1)}}\\ &\times\left(x-x_{ni1}^{(k-1)}\right)\Bigg)\end{aligned} \tag{5}$$

$$p\log(\epsilon+x) = \left(\frac{p}{\epsilon+x_{ni1}^{(k-1)}}\right)x + \text{const} \tag{6}$$

Thus, we can solve for $\vec{x}$ by initializing weights $\lambda_{ni}^{(0)} = 2p$ and then iteratively optimizing:

$$\vec{x}^{(k)} \leftarrow \operatorname{argmin}_{\vec{x}\in\mathcal{C}} \frac{1}{2}\|V(t)-\mathbf{W}\vec{x}\|_2^2 + \sum_{n,i}\lambda_{ni}^{(k-1)}x_{ni1} \tag{7}$$

$$\lambda_{ni}^{(k)} \leftarrow \frac{p}{\epsilon+x_{ni1}^{(k)}} \qquad \forall n,i \tag{8}$$

until convergence. Each iteration amounts to a convex optimization problem which can be solved efficiently. Upon convergence, the coefficient groups, $\{x_{nij}\}_{j=1,2,3}$, are converted into spike amplitudes and times (Ekanadham et al., 2011a).

1.1.3. Solving for spike waveforms: least squares

Having solved for a set of spike amplitudes and times using the procedure above, we then switch to the second half of the coordinate descent optimization procedure, refining the waveform shapes $W_n(t)$ while holding constant the estimated values for a_{ni} and τ_{ni}. Since the second term in Eq. (3) is a constant in this situation, the optimization is a least squares problem that can be solved efficiently using standard methods.

1.1.4. Initialization of waveform shapes

We initiate the coordinate descent algorithm by computing an initial estimate of the waveform shapes, $\{W_n(t)\}$. For this purpose, we used the centroids of clusters obtained using K-means clustering (see Section 1.2). We emphasize that the cluster assignments are *not* assumed to identify individual spikes – only the centroids are used, as initial waveform estimates, which are then used by the first CBP step of the coordinate descent algorithm in order to estimate spike amplitudes/times. For cells that produce a substantial number of isolated spikes, these initial waveform estimates are typically quite accurate, and we found that their subsequent refinement through iterative coordinate descent offered only minor improvement. However, in cases where the initial waveform estimates are noisy or incorrect, the iterative refinement of waveform shapes did offer significant improvement (e.g., see Fig. 5(c)). For one data set (not shown), we initialized the algorithm with a set

of random white noise waveforms, and verified that although this required substantially more computation (i.e., many coordinate descent iterations), the accuracy of the identified spikes was unaffected.

1.1.5. Improvements in computational efficiency

To improve computational efficiency, the voltage trace was partitioned into non-overlapping excerpts separated by intervals of silence, and each excerpt was processed independently. Silences were defined as intervals with duration longer than half the minimal waveform duration (approximately 2 ms) in which the voltage trace did not exceed the threshold in Eq. (9). Each excerpt was tested for whether it could be explained with a single isolated waveform shifted to an appropriate time. This was done by first identifying the best waveform n^* and time bin i^* using standard template matching, and then solving Eq. (4) with the restriction that only the $x_{n^*i^*j}$'s could be nonzero. If the energy of the residual obtained by subtracting the optimally placed waveform was less than a fixed percentile (99.999, chosen based on the spike waveform amplitudes relative to the noise level) of a chi-squared distribution (the distribution of the noise energy), then this single-waveform explanation for the interval was accepted. Otherwise, the interval was processed by solving the full CBP problem expressed in Eq. (4). We verified that this computational shortcut had negligible impact on the solution.

1.1.6. Thresholding of spike amplitudes

Since the spike amplitudes inferred from the solution of Eq. (4) can take on any nonnegative value, a threshold must be used for final spike identification. This threshold value determines the balance between missed spikes and false positives, which is a choice best left up to the investigator, who can assess the relative costs of the two types of error with regards to the scientific goals of the experiment. For the purposes of providing a simple automated choice, however, we compute a smoothed estimate (Gaussian kernel density estimator (Botev et al., 2010)) of the spike amplitude density and identify the largest value (before the peak) at which the density has a local minimum. The red vertical lines in Fig. 7 indicate this automatically computed threshold. Note that if there are multiple cells with similar-shaped waveforms but different amplitudes, the spike amplitude distribution will be multimodal, and multiple thresholds should be chosen.

1.2. Clustering method

For comparison purposes, we implemented a conventional three-step clustering method (Lewicki, 1998), as illustrated in Fig. 1, and used it to obtain all performance benchmarks presented in Section 2.

1.2.1. Detection

To identify segments of the voltage trace containing spiking activity, we identify peaks exceeding a threshold that is derived from an estimate of the noise level (Harris et al., 2000; Quiroga et al., 2004; Donoho and Johnstone, 1994):

$$T := 4\hat{\sigma} \qquad \hat{\sigma} := \frac{median(|V(t)|)}{0.6745} \tag{9}$$

Five millisecond windows, centered around these peaks, are identified as segments of spiking activity in the signal. The windowed segments are temporally upsampled by a factor of five using cubic spline interpolation, re-centered about the maximal value (across all electrodes) assuming zero padding on both ends, and then downsampled to the original rate.

1.2.2. Feature extraction

We then reduced the dimensionality of the data by projecting each of the segments into a low-dimensional feature space. Specifically, we formed a data vector for each segment (i.e., the elements are the voltage samples of all electrodes lying within the segment). We performed principal components analysis (PCA) on this set of vectors, selected the leading components that accounted for 90% of the variance over all the segments (e.g., see legends of Fig. 5b and c), and projected the contents of all segments onto these components. Fig. 4(a, c, and e) show the projections of these segments onto the first two principal components for three different data sets.

1.2.3. Clustering

The dimensionally reduced feature vectors are then automatically grouped according to similarity. For our primary comparisons we used K-means clustering to accomplish this (Duda et al., 2001). The number of clusters was manually adjusted to minimize the number of errors, and the procedure was run several times starting from randomly chosen initial values and the best of these solutions was selected. For the simulated data set, we also compare our results with those of a superparamagnetic clustering method (in the space of wavelet coefficients) (Quiroga et al., 2004). For this method, we obtain a lower bound on the number of errors by adding three numbers reported in the original paper: (1) the number of detection errors, (2) the number of classification errors, and (3) the number of detected voltage segments that contain two or more spikes (all clustering methods are guaranteed to miss at least one spike from such a segment).

1.3. Preprocessing of data

1.3.1. Filtering of raw voltage trace

All extracellular traces, both simulated and real, were highpass-filtered at 250Hz with a Butterworth filter of order 50. No preprocessing (other than amplitude re-scaling) was done for intracellular traces. For each tetrode data set, a sustained period in which the recording was stable was selected for analysis. For one of the tetrode sets (Harris et al., 2000), a period of 0.8 s containing anomalous bursting activity was removed from the analysis.

1.3.2. Ground truth spike identification from intracellular trace

For real data, ground truth spikes were inferred by identifying peaks in the intracellular traces that exceeded 4 standard deviations from the baseline. These traces have negligible noise, and this simple procedure can reliably and accurately identify all spikes (see Fig. 6).

1.3.3. Noise covariance estimation

We assume that the full spatio-temporal noise covariance matrix is space-time separable, allowing us to first whiten each channel in time, and then whiten across channels. For each channel, the temporal covariance matrix was assumed to be Toeplitz (i.e. stationary noise), depending only on the noise autocovariance. The autocovariance was estimated from "noise" regions in the extracellular trace which did not exceed $2\hat{\sigma}$ for a period of 50 ms or more, where $\hat{\sigma}$ was computed as in Eq. (9). A whitening filter was then computed by taking the central column of the inverse matrix-square-root of the temporal autocovariance matrix. Once each channel was whitened in time (by convolving with the whitening filter), the spatial covariance matrix (across electrodes) was estimated from the same noise regions. Each time slice was then left-multiplied by the inverse matrix-square-root.

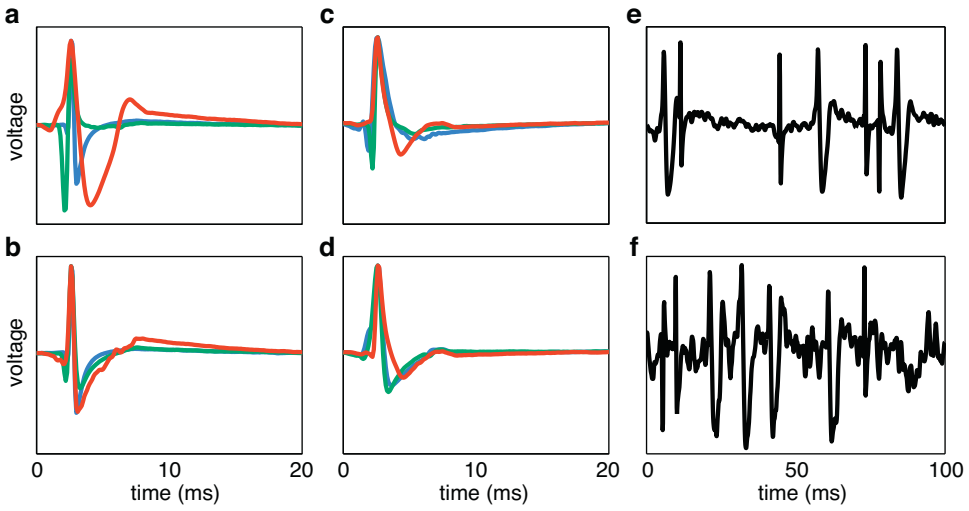

Fig. 3. Examples of simulated electrode data (Quiroga et al., 2004). (a–d) Each panel shows spike waveforms of three distinct cells, used to generate the simulated data sets. (e and f) Example simulated voltage traces using waveforms in (a), with noise levels $\sigma = 0.1$ and $\sigma = 0.2$, respectively.

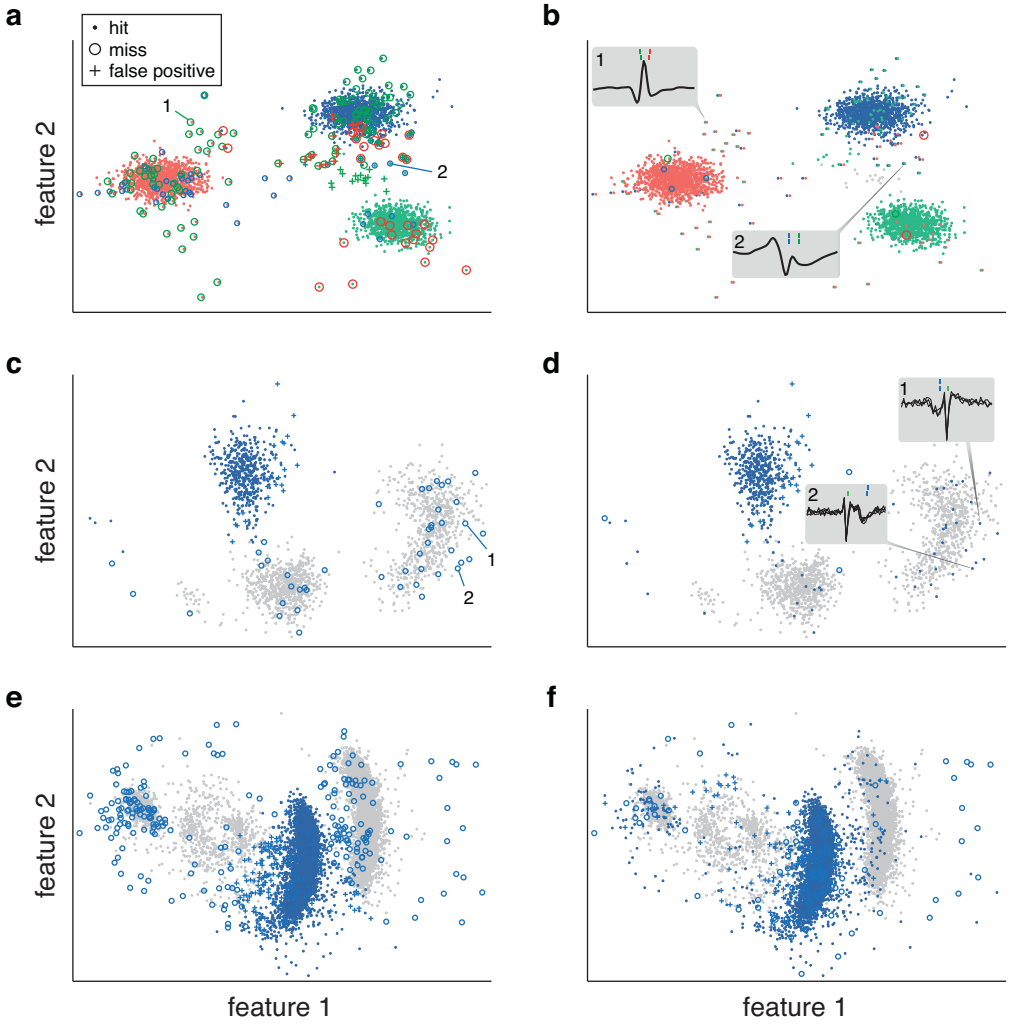

Fig. 4. Visualization of spike sorting results for simulated data using waveforms in Fig. 3(a) (top row), and tetrode data (Harris et al., 2000; Wehr et al., 1999) (middle and bottom rows, respectively). (a, c, and e) Spike identification arising from *K*-means clustering (Section 1.2). Each marker represents the projection of a voltage segment onto the leading 2 principal components. Points, circles, and crosses represent hits, missed spikes, and false positives, respectively, with color indicating cell identity. Gray points (bottom two rows) correspond to segments of real data for which no ground truth is available. (b, d, and f) Spike identification arising from our method. Insets show example voltage segments containing overlapping spikes (corresponding to numbered points in (a and c)) in the time domain. Colored vertical lines in the insets represent the occurrence times of ground truth spikes (top row) and spikes estimated by our method (bottom row).

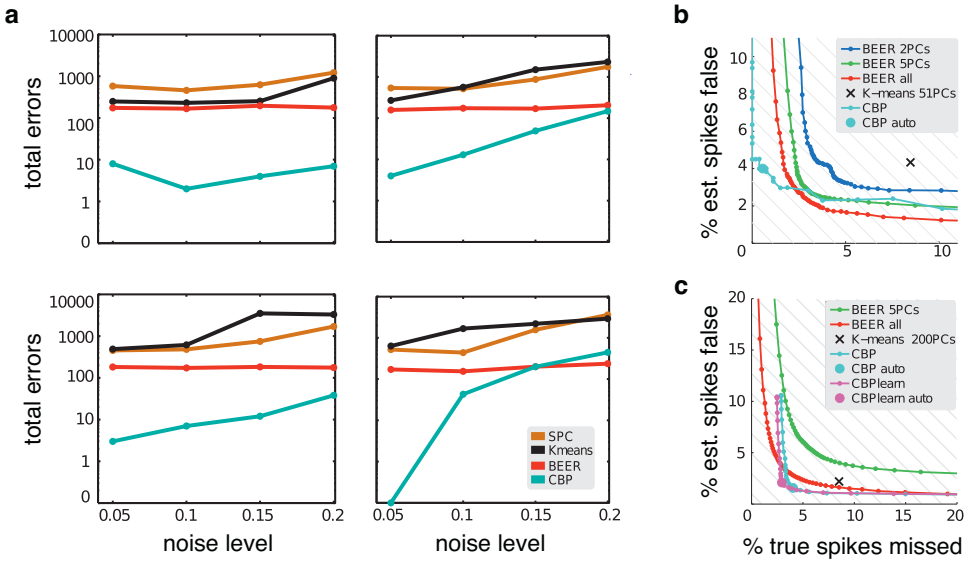

Fig. 5. Spike sorting performance comparison. (a) Each panel shows the total sorting errors (as a function of noise level) for each of the 4 simulated data sets (Quiroga et al., 2004) incurred by *K*-means clustering (black, Section 1.2), superparamagnetic clustering (Quiroga et al., 2004) (brown), BEER (red, Section 1.4), and our method (cyan, Section 1.1). For all four examples, a fixed threshold of 0.5 was used to identify spikes in our method. (b and c) Tradeoff between "false positive and "miss errors on each of the tetrode data sets (Harris et al., 2000; Wehr et al., 1999), respectively, as the assignment probability threshold is varied for the BEER (blue/green/red curves) and as the spike coefficient threshold is varied for our method (cyan curve, magenta curve with waveform learning). In (b), waveform learning did not significantly improve performance with CBP, and so the corresponding curve is not shown. Large points indicate results obtained with automatically chosen thresholds (see Section 1.1). The black X indicates the performance of *K*-means clustering (Section 1.2). Diagonal gray lines indicate contours of constant total error.

1.4. Performance evaluation

1.4.1. Counting misses and false positives

For evaluation, we matched spikes in the estimated spike train with spikes in the true spike train. An estimated spike was considered to match a true spike of the same cell if it occurred within 4 ms of the true spike time. True spikes with no matching estimated spike were categorized as "misses", and estimated spikes with no matching true spike were categorized as "false alarms".

1.4.2. Best ellipsoidal error rate (BEER)

We also computed the *best ellipsoidal error rate* (BEER) measure (Harris et al., 2000), which serves as an upper bound on the performance of any clustering-based spike sorting method that uses elliptical cluster boundaries. After thresholding and feature extraction, the windowed segments of the trace were labeled according to whether or not they contained a true spike. Half of this labeled data set was then used to train a support vector machine whose decision rule was a linear combination of all pairwise products of the features of each segment, and was thus capable of achieving any elliptical decision boundary. This decision boundary was then used to predict the occurrence of spikes in the segments in the remaining half of the labeled data, and the success or failure of these predictions then provided an estimate of the miss and false positive rate.

2. Results

2.1. Simulated data

We first apply our method to four simulated data sets (Quiroga et al., 2004), each containing spiking activity from three neurons (Poisson rate of 20 Hz, removing any spikes violating a 2 ms refractory period) with background noise at four different levels. Waveforms (shown in Fig. 3(a–d)) were taken from real recordings, and noise was constructed to reflect realistic background activity. Excerpts of voltage traces for two different noise levels are shown in Fig. 3(e and f).

The top row of Fig. 4 compares the spikes sorted by our method to those arising from *K*-means clustering for one of the data sets (corresponding waveforms shown in Fig. 3(a)), plotted in the space of the first two principal components. Although clustering correctly identifies many of the spikes, it misses a substantial subset that are distant from the cluster centers. Our method correctly recovers nearly all of these missed spikes. The inset graphs of Fig. 4(b) show voltage snippets corresponding to two such examples that do appear to contain superpositions of multiple spikes.

Fig. 5 (a) compares the total number of errors of our method with three other methods: (1) standard clustering using PCA and *K*-means (Section 1.2); (2) superparamagnetic clustering (Quiroga et al., 2004); (3) the *best ellipsoid error rate* (BEER) measure (Harris et al., 2000). Note that the BEER cannot be used as an actual spike sorting method – its parameters are adjusted to optimize performance on data for which the true spikes are known – but serves instead as a bound on the performance of *any* clustering-based method that uses elliptical boundaries (see Section 1.4 for details). Our method substantially outperforms all three other methods under all conditions, except for the highest noise level of the fourth data set (for which BEER shows slightly better performance).

2.2. Tetrode data from rat hippocampus (Harris et al., 2000)

We also applied our method to a portion of publicly available data, recorded from CA1 in anesthetized rat hippocampus (Harris et al., 2000). The data include simultaneous recordings from an extracellular tetrode, and an intracellular electrode that was used to obtain the actual spike times (so-called ground truth) for a single cell. Fig. 6 shows an excerpt of the tetrode recording, with the scaled intracellular trace superimposed in gray. There are two prominent waveforms appearing in the recording, the smaller of which corresponds to the intracellularly recorded cell.

Fig. 4(c) illustrates the performance for clustering (Section 1.2). Notice that, unlike the simulated data of Fig. 4(a), there are

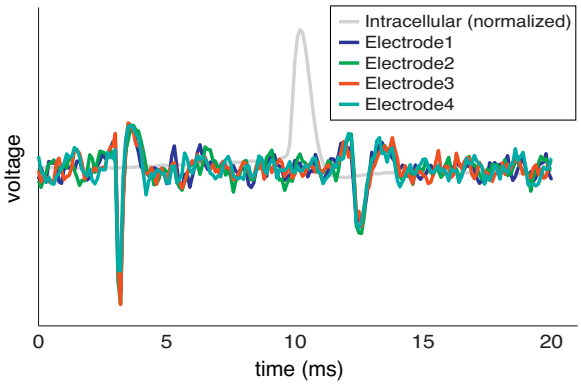

Fig. 6. Example portion of tetrode data (Harris et al., 2000). The four filtered extracellular electrode responses (colored traces) indicate two prominent spikes of different shape. Also shown is the intracellular response of a single cell (gray trace, vertically re-scaled for visualization) for a single cell, corresponding to the second of the extracellular events. The intracellular spikes are easily and unambiguously identifiable, and typically precede the extracellular spike waveform by approximately 2 ms.

many segments that presumably contain spikes but for which no ground truth is available (gray dots). For the intracellularly recorded cell, the majority of isolated spikes are correctly identified (blue dots), but a substantial number of other spikes are missed (blue circles) because they are far from the cluster center in the feature space. The majority of these missed spikes are recovered by our method, as illustrated in Fig. 4(d). The insets of Fig. 4(d) demonstrate that, as with the simulated data, these missed spikes typically overlap with the waveform of another spike.

Fig. 7 shows the distribution of spike amplitudes for three cells obtained by our method. Notice that all cells have a significant amount of low-amplitude activity. The isolation of a group of spike amplitudes relative to noise and the amplitudes of other cells with similar waveform shapes provides an informal indication that the activity originates from a single cell. For example the higher-amplitude modes in the first two distributions are clearly isolated and most likely correspond to the activity of distinct cells. The lower-amplitude modes in these two distributions are likely due to background spikes. On the other hand, the two modes of the third distribution are not well separated, and any choice of threshold is likely to result in a substantial number of errors. The final stage of our method uses a threshold to decide whether a given portion of the voltage trace contains a spike or not. Changes in the choice of threshold will trade off the number of false positives and misses. A simple automatic procedure can be used to select a threshold, but the correct choice ultimately depends on the relative cost of the two types of error, which can only be specified by the investigator.

Fig. 5(b) shows the tradeoff between misses and false positives incurred by *K*-means clustering, the BEER measure, and our method. We also tested the clustering method of (Harris et al., 2000) with a variety of parameter settings (not shown), and found its performance comparable to that of our benchmark clustering method. The error curves for the BEER measure were computed by varying the threshold on the class assignment probabilities computed with a quadratic classifier (see Section 1.4), and the error curves for our method were formed by varying the spike amplitude threshold.

2.3. Tetrode data in locust (Wehr et al., 1999)

We also applied our method to another data set, recorded from locust *in vivo* (Wehr et al., 1999). We applied the same analysis as in Section 2.2. Fig. 4(e and f) visualizes the spike sorting results for standard clustering and our method, while Fig. 5(c) compares our method's performance with that of standard clustering and the BEER. The ground truth cell is not as well isolated as that of the rat data set, resulting in a higher error rate for all methods. Despite this, our method again outperforms both clustering and the BEER measure, assuming equal weighting of misses and false positives. However, Fig. 4(f) indicates that our method misses some spikes that are correctly identified by clustering. We attribute this to the properties of the particular waveforms in this data set. Specifically, the voltage traces appear to contain spike waveforms of two neurons with very similar shapes, but different amplitudes. Our model allows spikes of any positive amplitude, but the objective function (Eq. 4) imposes a penalty that increases with amplitude. Therefore, our method prefers to explain a small-amplitude event in the voltage trace as a small-amplitude occurrence of a large waveform, rather than a large-amplitude occurrence of a similarly shaped small waveform. In the example of Fig. 4(f), the two neurons have slightly different waveform shapes, and the number of misses can be reduced slightly by iterating between refining the waveform shapes and estimating the spikes (see Section 1.1), as indicated by the magenta line in Fig. 5(c).

2.4. Algorithm computational complexity

The optimization of Eq. (4) was implemented in MATLAB using the CVX package (Grant and Boyd, 2010), a reasonably efficient and highly accurate package for convex optimization. We examined the computational costs of our algorithm as a function of three parameters: (1) the number of time samples in the voltage trace, (2) the number of distinct waveforms (neurons), and (3) the number of channels (electrodes). In practice, we can split the voltage

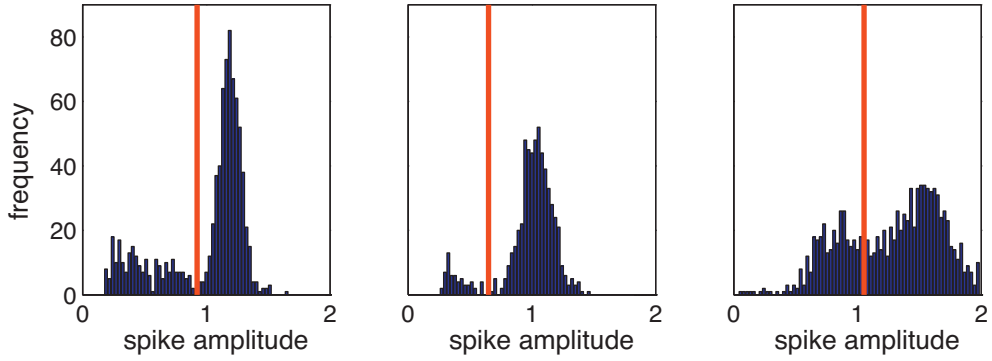

Fig. 7. Histograms of spike amplitudes for the three neurons corresponding to the waveforms estimated by our method, for tetrode data from (Harris et al., 2000). The middle plot corresponds to the cell for which ground truth is available via intracellular recordings. The red vertical line indicates an automatic threshold for identifying spikes, based on the procedure described in Section 1.1, and was used to obtain the error rates corresponding to the large cyan dot in Fig. 5(b).

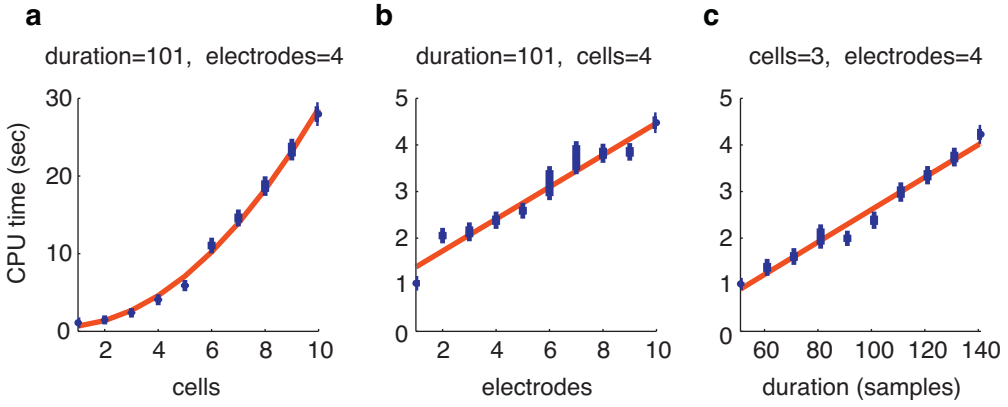

Fig. 8. The effect on computation time of three variables: the number of cells the number of electrodes and the number of voltage samples. Plotted values indicate mean CPU time required by the algorithm, running in Matlab on a conventional desktop computer, averaged over 25 trials. Error bars indicate standard error. The red line indicates a quadratic fit (a) or a linear fit (b and c).

trace $V(t)$ whenever there is a period without any spiking activity, and process the portions between these silences independently. As such, the first parameter is specified not in terms of the experiment length, but rather in terms of the typical duration between silences, which depends only on the firing rates. Fig. 8 shows the execution time for the algorithm as a function of each of these parameters, while keeping the other two fixed at typical values. The computation time grows approximately quadratically with the number of cells, and linearly with the temporal duration and number of electrodes. The last of these implies that the algorithm scales efficiently for multi-electrode arrays.

3. Discussion

Starting from a simple probabilistic model for extracellular voltage measurements, we have developed a unified sparse estimation methodology for spike sorting that iteratively optimizes both the waveform shapes and their respective spikes. We have shown that, on simulated and real data sets taken from recent literature, this method is much more accurate than clustering-based methods for resolving overlapping spikes (and equally accurate when resolving isolated spikes). By comparing performance with the BEER bound, we have also shown that our method outperforms the *entire class of clustering methods* that use elliptical boundaries, which encompasses the vast majority of spike sorting methods currently in use. Finally, our method is well-suited for sorting multiple cells and/or multiple electrodes, as it is reasonably efficient and requires minimal human intervention except for selecting the number of cells. Previous methods have estimated the number of cells based on probabilistic models (Lewicki, 1994; Wood and Black, 2008), and such techniques could be incorporated into our method. However, choosing the number of cells (and specifically, deciding whether to keep or discard the imperfectly estimated spikes of partially isolated neurons), is a decision that depends on the cost of errors, and so is perhaps best left to the investigator.

Despite the strong performance of the method, we see several opportunities for improvement. First, the current algorithm assumes that all spike waveforms can be differentiated by their shape, and cannot handle multiple spike waveforms that have the same shape but different amplitude (see Fig. 4(f), and Section 2.3). A more general solution should separate waveforms according to both their shape *and* amplitude similarity (for example, by including a post-processing step that is able to identify and partition multimodal amplitude distributions, and by replacing the sparsity term in the objective function with one that more directly encourages binary amplitude values). Second, our formulation assumes that waveform shapes remain constant throughout a recording session. However, it is well-known that tissue relaxation, electrode drift, and bursting activity, amongst other factors, can cause the waveform amplitude or shape associated with a cell to change over time. In these cases, our method will likely attempt to explain changes in shape using some linear combination of the original waveform along with low-amplitude spikes of other waveforms. If the change in shape is small relative to the noise, and is still closer to the original waveform than to the waveform of any other cell (or linear combination of cells), then the thresholding post-processing step is likely to identify the correct waveform. For slow drifts in waveform shape, one could apply the method separately to temporally restricted portions of the data, re-estimating the waveform shapes for each (Franke et al., 2009). A more elegant solution would be to develop an incremental version of the algorithm that operates on voltage measurements as they arrive, rather than operating simultaneously on the full data set. This would greatly reduce the memory requirements of the method, and would allow real-time sorting for use in interactive or adaptive experiments. Ultimately, a unified solution requires a more direct incorporation of the potential changes in shape or amplitude into the probabilistic model. For example, the waveform shapes could be estimated as a function of time by modeling the dynamics as a stationary process using an autoregressive (Chen et al., 2011) or Markov chain (Pouzat et al., 2004; Calabrese and Paninski, 2011) model.

Although the probabilistic framework underlying this work has been described in a number of previous publications (Sahani, 1999; Pillow et al., 2008, 2013), very few spike sorting methods make direct use of it, primarily because of the difficulty of solving the sparse linear inverse problem. Instead, most spike sorting methods are implemented as a sequence of procedural steps, with the central step a form of clustering that cannot handle spike superpositions, and with each step relying on additional free parameters and/or substantial human supervision, and introducing additional sources of error. By overcoming some of the technical challenges associated with spike inference, our results demonstrate the potential advantages of a unified probabilistic framework, providing a base on which future spike sorting methods may be built, and facilitating the objective comparison of their performance.

Acknowledgements

We thank Jonathan Pillow, Jonathon Shlens, and Peter Li for helpful discussions.

References

Atiya A. Recognition of multiunit neural signals. IEEE Transactions on Biomedical Engineering 1992;39(7):723–9.

Bar-Gad I, Ritov Y, Vaadia E, Bergman H. Failure in identification of overlapping spikes from multiple neuron activity causes artificial correlations. Journal of Neuroscience Methods 2001;107(1–2):1–13.

Botev ZI, Grotowski JF, Kroese DP. Kernel density estimation via diffusion. Annals of Statistics 2010;38(5):2916–57.

Brown EN, Kass RE, Mitra PP. Multiple neural spike train data analysis: state-of-the-art and future challenges. Nature Neuroscience 2004;7(5):456–61.

Calabrese A, Paninski L. Kalman filter mixture model for spike sorting of non-stationary data. Journal of Neuroscience Methods 2011;196(1):159–69.

Candes EJ. Enhancing sparsity by reweighted L_1 minimization. Journal of Fourier Analysis and Applications 2008;14:877–905.

Chen B, Carlson D, Carin L. On the analysis of multi-channel neural spike data. In: Adv. Neural InformationProcessing Systems (NIPS*11), vol. 24, MIT Press, Cambridge, MA, 2011.

Choi JH, Jung HK, Kim T. A new action potential detector using the MTEO and its effects on spike sorting systems at low signal-to-noise ratios. IEEE Transactions on Biomedical Engineering 2006;53(4):738–46.

Devries S. Correlated firing in rabbit retinal ganglion cells. Journal of Neurophysiology 1999;81(2):908–20.

Donoho DL, Johnstone IM. Ideal spatial adaptation by wavelet shrinkage. Biometrika 1994;81(3):425–55.

Duda RO, Hart PE, Stork DG. Pattern classification. 2nd ed. New York: Wiley-Interscience; 2001.

Ekanadham C, Tranchina D, Simoncelli EP. Recovery of sparse translation-invariant signals with continuous basis pursuit. IEEE Transactions on Signal Processing 2011a;59(10):4735–44.

Ekanadham C, Tranchina D, Simoncelli EP. A blind sparse deconvolution method for neural spike identification. In: Adv Neural Information Processing Systems (NIPS*11), vol. 24, MIT Press, Cambridge, MA, 2011b, Neural Information Processing Systems 24; 2011b.

Fee MS, Mitra PP, Kleinfeld D. Automatic sorting of multiple unit neuronal signals in the presence of anisotropic and non-Gaussian variability. Journal of Neuroscience Methods 1996;69(2):175–88.

Franke F, Natora M, Boucsein C, Munk M, Obermayer K. An online spike detection and spike classification algorithm capable of instantaneous resolution of overlapping spikes. Journal of Computational Neuroscience 2009:127–48.

Gerstein GL, Clark WA. Simultaneous studies of firing patterns in several neurons. Science 1964;143(3612):1325–7.

Grant M, Boyd S. CVX: Matlab software for disciplined convex programming, version 1.21; 2010 http://cvxr.com/cvx

Harris KD, Henze DA, Csicsvari J, Hirase H, Henze KDDA, Csicsvari J. Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements. Journal of Neurophysiology 2000;84:401–14.

Kwon KY, Oweiss K. Wavelet footprints for detection and sorting of extracellular neural action potentials. In: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2011. p. 609–12.

Lewicki M. Bayesian modeling and classification of neural signals. Neural Computation 1994;6:1005–30.

Lewicki MS. A review of methods for spike sorting: the detection and classification of neural action potentials. Network 1998;9(4):R53–78.

Mastronarde D. Correlated firing of cat retinal ganglion cells. Journal of Neurophysiology 1989;49(2):303–24.

Meister M, Pine J, Baylor DA. Multi-neuronal signals from the retina: acquisition and analysis. Journal of Neuroscience Methods 1994;51(1):95–106.

Obeid I, Wolf P. Evaluation of spike-detection algorithms for a brain–machine interface application. IEEE Transactions on Biomedical Engineering 2004;51(6):905–11.

Pazienti A, Grn S. Robustness of the significance of spike synchrony with respect to sorting errors. Journal of Computational Neuroscience 2006;21:329–42.

Pillow JW, Shlens J, Paninski L, Sher A, Litke AM, Chichilnisky EJ, et al. Spatio-temporal correlations and visual signaling in a complete neuronal population. Nature 2008;454(7206):995–9.

Pillow J, Shlens J, Chichilsnisky E, Simoncelli EP. A model-based spike sorting algorithm for reducing correlation artifacts in multi-neuron recordings. PLoS ONE 2013;8(5)., http://dx.doi.org/10.1371/journal.pone.0062123.

Prentice JS, Homann J, Simmons KD, Tkaik G, Balasubramanian V, Nelson PC. Fast, scalable, Bayesian spike identification for multi-electrode arrays. PLoS ONE 2011;6(7):e19884.

Pouzat C, Delescluse M, Viot P, Diebolt J. Improved spike-sorting by modeling firing statistics and burst-dependent spike amplitude attenuation: a Markov chain Monte Carlo approach. Journal of Neurophysiology 2004;91(6):2910–28.

Quiroga RQ, Nadasdy Z, Ben-Shaul Y. Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. Neural Computation 2004;16:1661–87.

Rebrik SP, Wright BD, Emondi AA, Miller KD. Cross-channel correlations in tetrode recordings: implications for spike-sorting. Neurocomputing 1999;26–27(0):1033–8.

Rodieck RW. Maintained activity of cat retinal ganglion cells. Journal of Neurophysiology 1967;30(5):1043–71.

Rutishauser U, Schuman EM, Mamelak AN. Online detection and sorting of extracellularly recorded action potentials in human medial temporal lobe recordings, in vivo. Journal of Neuroscience Methods 2006;154(1–2):204–24.

Sahani M, Pezaris JS, Andersen RA. On the separation of signals from neighboring cells in tetrode recordings. In: NIPS; 1997.

Sahani M. Latent variable models for neural data analysis. Ph.D. thesis. Pasadena, CA: California Institute of Technology; 1999.

Schnitzer MJ, Meister M. Multineuronal firing patterns in the signal from eye to brain. Neuron 2003;37:499–511.

Segev R, Goodhouse J, Puchalla J, Berry MJ. Recording spikes from a large fraction of the ganglion cells in a retinal patch. Nature Neuroscience 2004;7(10):1154–61.

Shlens J, Rieke F, Chichilnisky E. Synchronized firing in the retina. Current Opinion in Neurobiology 2008;18(4):396–402.

Shlens J, Field GD, Gauthier JL, Greschner M, Sher A, Litke AM, et al. The structure of large-scale synchronized firing in primate retina. Journal of Neuroscience 2009;29:5022–31.

Shoham S, Fellows MR, Normann RA. Robust, automatic spike sorting using mixtures of multivariate *t*-distributions. Journal of Neuroscience Methods 2003;127(2):111–22.

Takahashi S, Anzai Y, Sakurai Y. Automatic sorting for multi-neuronal activity recorded with tetrodes in the presence of overlapping spikes. Journal of Neurophysiology 2003;89(4):2245–58.

Takahashi S, Sakurai Y. Real-time and automatic sorting of multi-neuronal activity for sub-millisecond interactions in vivo. Neuroscience 2005;134(1):301–15.

Vargas-Irwin C, Donoghue JP. Automated spike sorting using density grid contour clustering and subtractive waveform decomposition. Journal of Neuroscience Methods 2007;164(1):1–18.

Wehr M, Pezaris JS, Sahani M. Simultaneous paired intracellular and tetrode recordings for evaluating the performance of spike sorting algorithms. Neurocomputing 1999;26–27:1061–8.

Wood F, Black MJ, Vargas-irwin C, Fellows M, Donoghue JP. On the variability of manual spike sorting. IEEE Transactions on Biomedical Engineering 2004;51:912–8.

Wood F, Black MJ. A non-parametric Bayesian alternative to spike sorting. Journal of Neuroscience Methods 2008;173:1–12.

Zhang P-M, Wu J-Y, Zhou Y, Liang P-J, Yuan J-Q. Spike sorting based on automatic template reconstruction with a partial solution to the overlapping problem. Journal of Neuroscience Methods 2004;135(1–2):55–65.