LETTER Communicated by Maneesh Sahani

Unsupervised Spike Detection and Sorting with Wavelets and Superparamagnetic Clustering

R. Quian Quiroga
rodri@vis.caltech.edu
Z. Nadasdy
zoltan@vis.caltech.edu
Division of Biology, California Institute of Technology, Pasadena, CA 91125, U.S.A.

Y. Ben-Shaul
ybss@md.huji.ac.il
ICNC, Hebrew University, Jerusalem, Israel

This study introduces a new method for detecting and sorting spikes from multiunit recordings. The method combines the wavelet transform, which localizes distinctive spike features, with superparamagnetic clustering, which allows automatic classification of the data without assumptions such as low variance or gaussian distributions. Moreover, an improved method for setting amplitude thresholds for spike detection is proposed. We describe several criteria for implementation that render the algorithm unsupervised and fast. The algorithm is compared to other conventional methods using several simulated data sets whose characteristics closely resemble those of in vivo recordings. For these data sets, we found that the proposed algorithm outperformed conventional methods.

1 Introduction

Many questions in neuroscience depend on the analysis of neuronal spiking activity recorded under various behavioral conditions. For this reason, data acquired simultaneously from multiple neurons are invaluable for elucidating principles of neural information processing. Recent advances in commercially available acquisition systems allow recordings of up to hundreds of channels simultaneously, and the reliability of these data critically depends on accurately identifying the activity of individual neurons. However, developments of efficient and reliable computational methods for classifying multiunit data, that is, spike sorting algorithms, lag behind the capabilities afforded by current hardware. In practice, supervised spike sorting of a large number of channels is highly time-consuming and nearly impossible to perform during the course of an experiment.

The basic algorithmic steps of spike classification are as follows: (1) spike detection, (2) extraction of distinctive features from the spike shapes, and (3) clustering of the spikes by these features. Spike sorting methods are typically based on clustering predefined spike shape features such as peak-to-peak amplitude, width, or principal components (Abeles & Goldstein, 1977; Lewicki, 1998). Nevertheless, it is impossible to know beforehand which of these features is optimal for discriminating between spike classes in a given data set. In the specific case where the spike features are projections on the first few principal components, the planes onto which the spikes are projected maximize the variance of data but do not necessarily provide an optimal separation between the clusters. A second critical issue is that even when optimal features from a given data set are used for classification, the distribution of the data imposes additional constraints on the clustering procedure. In particular, violation of normality in a given feature's distribution compromises most unsupervised clustering algorithms, and therefore manual clustering of the data is usually preferred. However, besides being a very time-consuming task, manual clustering introduces errors due to both the limited dimensionality of the cluster cutting space and human biases (Harris, Henze, Csicsvari, Hirase, & Buzsáki, 2000). An alternative approach is to define spike classes by a set of manually selected thresholds (window discriminators) or with spike templates. Although this is computationally very efficient and can be implemented on-line, it is reliable only when the signal-to-noise ratio is very high and it is limited to the number of channels a human operator is able to supervise.

In this article, we introduce a new method that improves spike separation in the feature space and implements a novel unsupervised clustering algorithm. Combining these two features results in a novel unsupervised spike sorting system. The cornerstones of our method are the wavelet transform, which is a time-frequency decomposition of the signal with optimal resolution in both the time and the frequency domains, and superparamagnetic clustering (SPC; Blatt, Wiseman, & Domany, 1996), a relatively new clustering procedure developed in the context of statistical mechanics. The complete algorithm encompasses three principal stages: (1) spike detection, (2) selection of spike features, and (3) clustering of the selected spike features.

In the first step, spikes are detected with an automatic amplitude threshold on the high-pass filtered data. In the second step, a small set of wavelet coefficients from each spike is chosen as input for the clustering algorithm. Finally, the SPC classifies the spikes according to the selected set of wavelet coefficients. We stress that the entire process of detection, feature extraction, and clustering is performed without supervision and relatively quickly. In this study, we compare the performance of the algorithm with other methods using simulated data that closely resemble real recordings. The rationale of using simulated data is to obtain an objective measure of performance, since the simulation sets the identity of the spikes.

2 Theoretical Background

2.1 Wavelet Transform. The wavelet transform (WT) is a time-frequency representation of the signal that has two main advantages over conventional methods: it provides an optimal resolution in both the time and the frequency domains, and it eliminates the requirement of signal stationarity. It is defined as the convolution between the signal $x(t)$ and the wavelet functions $\psi_{a,b}(t)$,

$$W_\psi X(a, b) = \langle x(t) \mid \psi_{a,b}(t) \rangle, \tag{2.1}$$

where $\psi_{a,b}(t)$ are dilated (contracted), and shifted versions of a unique wavelet function $\psi(t)$,

$$\psi_{a,b}(t) = |a|^{-\frac{1}{2}} \psi \left(\frac{t-b}{a} \right) \tag{2.2}$$

where a and b are the scale and translation parameters, respectively. Equation 2.1 can be inverted, thus providing the reconstruction of $x(t)$.

The WT maps the signal that is represented by one independent variable t onto a function of two independent variables a, b. This procedure is redundant and inefficient for algorithmic implementations; therefore, the WT is usually defined at discrete scales a and discrete times b by choosing the set of parameters $\{a_j = 2^{-j}; b_{j,k} = 2^{-j}k\}$, with integers j and k. Contracted versions of the wavelet function match the high-frequency components, while dilated versions match the low-frequency components. Then, by correlating the original signal with wavelet functions of different sizes, we can obtain details of the signal at several scales. These correlations with the different wavelet functions can be arranged in a hierarchical scheme called multiresolution decomposition (Mallat, 1989). The multiresolution decomposition algorithm separates the signal into details at different scales and a coarser representation of the signal named "approximation" (for details, see Mallat, 1989; Chui, 1992; Samar, Swartz, & Raghveer, 1995; Quian Quiroga, Sakowicz, Basar, & Schürmann, 2001; Quian Quiroga & Garcia, 2003).

In this study we implemented a four-level decomposition using Haar wavelets, which are rescaled square functions. Haar wavelets were chosen due to their compact support and orthogonality, which allows the discriminative features of the spikes to be expressed with a few wavelet coefficients and without a priori assumptions on the spike shapes.

2.2 Superparamagnetic Clustering. The following is a brief description of the key ideas of superparamagnetic clustering (SPC), which is based on simulated interactions between each data point and its K-nearest neighbors (for details, see Blatt et al., 1996; Blatt, Wiseman, & Domany, 1997). The method is implemented as a Monte Carlo iteration of a Potts model. The

Potts model is a generalization of the Ising model where instead of having spins with values $\pm 1/2$, there are q different states per particle (Binder & Heermann, 1988).

The first step is to represent the m selected features of each spike i by a point $\boldsymbol{x}_i$ in an m-dimensional phase space. The interaction strength between points $\boldsymbol{x}_i$ is then defined as

$$J_{ij} = \begin{cases} \frac{1}{K} \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2a^2}\right) & \text{if } \mathbf{x}_i \text{ is a nearest neighbor of } \mathbf{x}_j, \\ 0 & \text{else} \end{cases} \tag{2.3}$$

where a is the average nearest-neighbors distance and K is the number of nearest neighbors. Note that the strength of interaction J_{ij} between nearest-neighbor spikes falls off exponentially with increasing Euclidean distance $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|^2$, which corresponds to the similarity of the selected features (i.e., similar spikes will have a strong interaction).

In the second step, an initial random state s from 1 to q is assigned to each point $\boldsymbol{x}_i$. Then N Monte Carlo iterations are run for different temperatures T using the Wolf algorithm (Wolf, 1989; Binder & Heermann, 1988). Blatt et al. (1997) used a Swendnsen-Wang algorithm instead, but its implementation and performance are both very similar. The advantage of both algorithms over simpler approaches such as the Metropolis algorithm is their enhanced performance in the superparamagnetic regime (see Binder & Heermann, 1988; Blatt et al., 1997, for details). The main idea of the Wolf algorithm is that given an initial configuration of states s, a point $\boldsymbol{x}_i$ is randomly selected and its state s changed to a new state s_{new}, randomly chosen between 1 and q. The probability that the nearest neighbors of $\boldsymbol{x}_i$ will also change their state to s_{new} is given by

$$p_{ij} = 1 - \exp\left(-\frac{J_{ij}}{T}\delta_{s_i,s_j}\right), \tag{2.4}$$

where T is the temperature (see below). Note that only those nearest neighbors of $\boldsymbol{x}_i$ that were in the same previous state s are the candidates to change their values to s_{new}. Neighbors that change their values create a "frontier" and cannot change their value again during the same iteration. Points that do not change their value in a first attempt can do so if revisited during the same iteration. Then for each point of the frontier, we apply equation 2.4 again to calculate the probability of changing the state to s_{new} for their respective neighbors. The frontier is updated, and the update is repeated until the frontier does not change any more. At that stage, we start the procedure again from another point and repeat it several times in order to get representative statistics. Points that are relatively close together (i.e., corresponding to a given cluster) will change their state together. This observation can be quantified by measuring the point-point correlation $\langle \delta_{s_i,s_j} \rangle$ and defining $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ to be members of the same cluster if $\langle \delta_{s_i,s_j} \rangle \geq \theta$, for a given threshold θ.

As in Blatt et al. (1996), we used $q = 20$ states, $K = 11$ nearest neighbors, $N = 500$ iterations, and $\theta = 0.5$. It has indeed been shown that clustering results mainly depend on the temperature and are robust to small changes in the previous parameters (Blatt et al., 1996).

Let us now discuss the role of the temperature T. Note from equation 2.4 that high temperatures correspond to a low probability of changing the state of neighboring points together, whereas low temperatures correspond to a higher probability regardless of how weak the interaction J_{ij} is. This has a physical analogy with a spin glass, in which at a relatively high temperature, all the spins are switching randomly, regardless of their interactions (paramagnetic phase). At a low temperature, the entire spin glass changes its state together (ferromagnetic phase). However, at a certain medium range of temperatures, the system reaches a "superparamagnetic" phase in which only those spins that are grouped together will change their state simultaneously. Regarding our clustering problem, at low temperatures, all points will change their state together and will therefore be considered as a single cluster; at high temperatures, many points will change their state independently from one another, thus partitioning the data into several clusters with only a few points in each; and for temperatures corresponding to the superparamagnetic phase, only those points that are grouped together will change their state simultaneously.

Figure 1A shows a two-dimensional (2D) example in which 2400 2D points were distributed in three different clusters. Note that the clusters partially overlap, they have a large variance, and, moreover, the centers fall outside the clusters. In particular, the distance between arbitrarily chosen points of the same cluster can be much larger than the distance between points in different clusters. These features render the use of conventional clustering algorithms unreliable. The different markers represent the outcome after clustering with SPC. Clearly, most of the points were correctly classified. In fact, only 102 of 2400 (4%) data points were not classified because they were near the boundaries of the clusters. Figure 1B shows the number of elements assigned to each given cluster as a function of the temperature. At low temperatures, we have a single cluster with all 2400 data points included. At a temperature between 0.04 and 0.05, this cluster breaks down into three subclusters corresponding to the superparamagnetic transition. The classification shown in the upper plot was performed at $T = 0.05$. At about $T = 0.08$, we observe the transition to the paramagnetic phase, where the clusters break down into several groups with a few members each.

Note that the algorithm is based on K-nearest neighbor interactions and therefore does not assume that clusters are nonoverlapping or that they have low variance or a gaussian distribution.

3 Description of the Method

Figure 2 summarizes the three principal stages of the algorithm: (1) spikes are detected automatically via amplitude thresholding; (2) the wavelet trans-

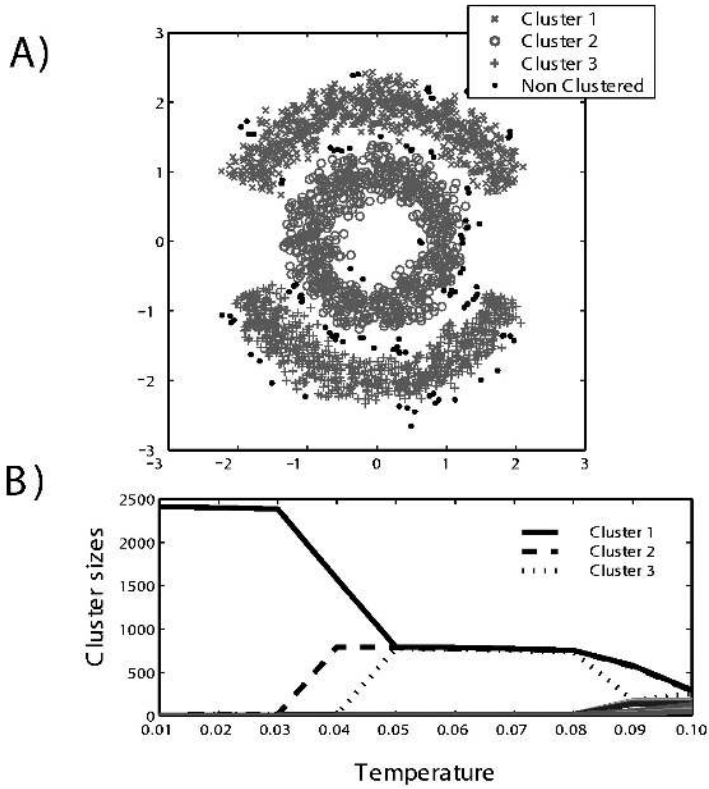

Figure 1: Example showing the performance of superparamagnetic clustering. (A) The two-dimensional data points used as inputs. The different markers represent the outcome of the clustering algorithm. (B) Cluster size vs. temperature. At temperature 0.05, the transition to the superparamagnetic phase occurs, and the three clusters are separated.

form is calculated for each of the spikes and the optimal coefficients for separating the spike classes are automatically selected; and (3) the selected wavelet coefficients then serve as the input to the SPC algorithm, and clustering is performed after automatic selection of the temperature corresponding to the superparamagnetic phase. (A Matlab implementation of the algorithm can be obtained on-line from www.vis.caltech.edu/∼rodri.)

3.1 Spike Detection. Spike detection was performed by amplitude thresholding after bandpass filtering the signal (300–6000 Hz, four pole

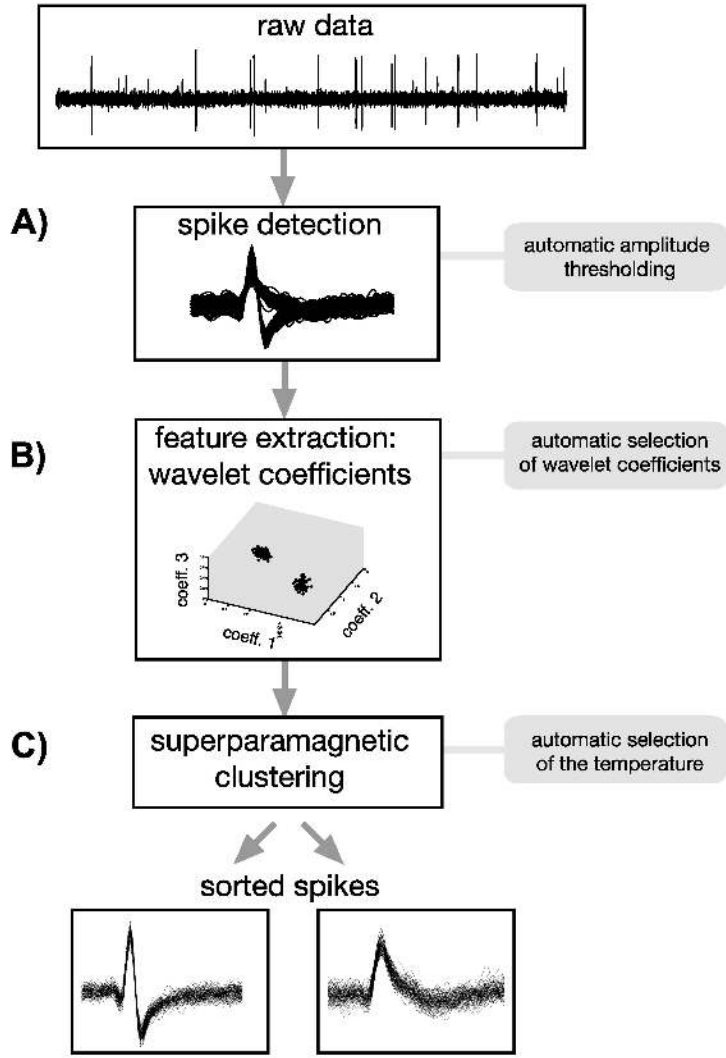

Figure 2: Overview of the automatic clustering procedure. (A) Spikes are detected by setting an amplitude threshold. (B) A set of wavelet coefficients representing the relevant features of the spikes is selected. (C) The SPC algorithm is used to cluster the spikes automatically.

butterworth filter). The threshold (*Thr*) was automatically set to

$$Thr = 4\sigma_n; \qquad \sigma_n = median\left\{\frac{|x|}{0.6745}\right\}, \tag{3.1}$$

where x is the bandpass-filtered signal and σ_n is an estimate of the standard deviation of the background noise (Donoho & Johnstone, 1994). Note that taking the standard deviation of the signal (including the spikes) could lead to very high threshold values, especially in cases with high firing rates and large spike amplitudes. In contrast, by using the estimation based on the median, the interference of the spikes is diminished (under the reasonable assumption that spikes amount to a small fraction of all samples). To demonstrate this, we generated a segment of 10 sec of background noise with unit standard deviation, and in successive simulations, we added a distinct spike class with different firing rates. Figure 3 shows that for noise alone (i.e., zero firing rate), both estimates are equal, but as the firing rate increases, the standard deviation of the signal (conventional estimate) gives an increasingly erroneous estimate of the noise level, whereas the improved estimate from equation 3.1 remains close to the real value.

For each detected spike, 64 samples (i.e., ~2.5 ms) were saved for further analysis. All spikes were aligned to their maximum at data point 20. In order to avoid spike misalignments due to low sampling, spike maxima were determined from interpolated waveforms of 256 samples, using cubic splines.

3.2 Selection of Wavelet Coefficients. After spikes are detected, their wavelet transform is calculated, thus obtaining 64 wavelet coefficients for each spike. We implemented a four-level multiresolution decomposition using Haar wavelets. As explained in section 2.1, each wavelet coefficient characterizes the spike shapes at different scales and times. The goal is to select a few coefficients that best separate the different spike classes. Clearly, such coefficients should have a multimodal distribution (unless there is only one spike class). To perform this selection automatically, the Lilliefors modification of a Kolmogorov-Smirnov (KS) test for normality was used (Press, Teukolsky, Vetterling, & Flannery, 1992). Note that we do not rely on any particular distribution of the data; rather, we are interested in deviation from normality as a sign of a multimodal distribution. Given a data set x, the test compares the cumulative distribution function of the data ($F(x)$) with that of a gaussian distribution with the same mean and variance ($G(x)$). Deviation from normality is then quantified by

$$\max(|F(x) - G(x)|). \tag{3.2}$$

In our implementation, the first 10 coefficients with the largest deviation from normality were used. The selected set of wavelet coefficients provides

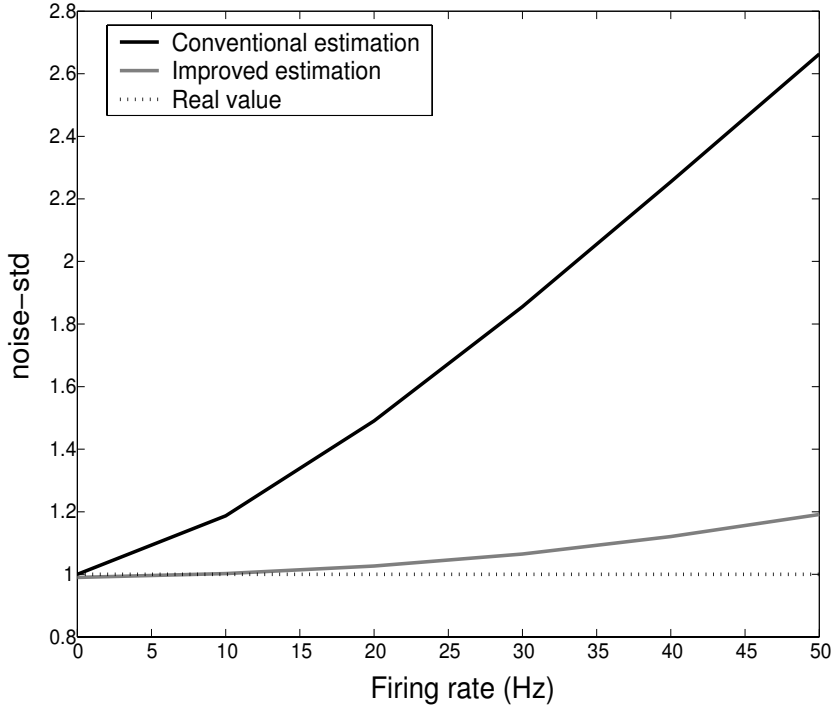

Figure 3: Estimation of noise level used for determining the amplitude threshold. Note how the conventional estimation based on the standard deviation of the signal increases with the firing rate, whereas the improved estimation from equation 3.1 remains close to the real value. See the text for details.

a compressed representation of the spike features that serves as the input to the clustering algorithm.

Overlapping spikes (i.e., spikes from different neurons appearing quasi-simultaneously) introduce outliers in the distribution of the wavelet coefficients that cause deviations from normality in unimodal (as well as multimodal) distributions, thus compromising the use of the KS test as an estimation of multimodality. In order to minimize this effect, for each coefficient we only considered values within ± 3 standard deviations.

3.3 SPC and Localization of the Superparamagnetic Phase. Once the selected set of wavelet coefficients is chosen, we run the SPC algorithm for a wide range of temperatures spanning the ferromagnetic, superparamag-

netic, and paramagnetic phases. In order to localize the superparamagnetic phase automatically, a criterion based on the cluster sizes is used. The idea is that for both the paramagnetic and ferromagnetic phases, temperature increases can only lead to the creation of clusters with few members each. Indeed, in the paramagnetic phase (i.e., high temperature), the clusters break down into several small ones, and in the ferromagnetic phase, there are almost no changes when the temperature is increased. In contrast, in the superparamagnetic phase, increasing the temperature creates new clusters with a large number of members.

In our implementation, we varied the temperature from 0 to 0.2 in increments of 0.01 and looked for the highest temperature at which a cluster containing more than 60 points appeared (not being present at lower temperatures). Since our simulations were 60 sec long, this means that we considered clusters corresponding to neurons with a mean firing rate of at least 1 Hz. The threshold of 1 Hz gave us optimal results for all our simulations, but it should be decreased if one considers neurons with lower firing rates. Alternatively, one can consider a fraction of the total number of spikes. If no cluster with a minimum of 60 points was found, we kept the minimum temperature value. Using this criterion, we can automatically select the optimal temperature for cluster assignments, and therefore the whole clustering procedure becomes unsupervised.

4 Data Simulation

Simulated signals were constructed using a database of 594 different average spike shapes compiled from recordings in the neocortex and basal ganglia. For generating background noise, spikes randomly selected from the database were superimposed at random times and amplitudes. This was done for half the times of the samples. The rationale was to mimic the background noise of actual recordings that is generated by the activity of distant neurons. Next, we superimposed a train of three distinct spike shapes (also preselected from the same database of spikes) on the noise signal at random times. The amplitude of the three spike classes was normalized to have a peak value of 1. The noise level was determined from its standard deviation, which was equal to 0.05, 0.1, 0.15, and 0.2 relative to the amplitude of the spike classes. In one case, since clustering was relatively easy, we also considered noise levels 0.25, 0.30, 0.35, and 0.4. Spike times and identities were saved for subsequent evaluation of the clustering algorithm. The data were first simulated at a sampling rate of 96,000 Hz, and by using interpolated waveforms of the original spike shapes, we simulated the spike times to fall continuously between samples (to machine precision). Finally, the data were downsampled to 24,000 Hz. This procedure was introduced in order to imitate actual recording conditions in which samples do not necessarily fall on the same features within a spike (i.e., the peak of the signal does not necessarily coincide with a discrete sample).

In all simulations, the three distinct spikes had a Poisson distribution of interspike intervals with a mean firing rate of 20 Hz. A 2 ms refractory period between spikes of the same class was introduced. Note that the background noise reproduces spike shape variability in biological signals (Fee, Mitra, & Kleinfeld, 1996; Pouzat, Mazor, & Laurent, 2002). Moreover, constructing noise from spikes ensures that this noise shares a similar power spectrum with the spikes themselves (1/f spectrum). The realistic simulation conditions applied here render the entire procedure of spike sorting more challenging than, for example, assuming a white noise distribution of background activity. Further complications of real recordings (e.g., overlapping spikes, bursting activity, moving electrodes) will be addressed in the next section.

Figure 4 shows one of the simulated data sets with a noise level 0.1. Figure 4A discloses the three spike shapes that were added to the background noise, as shown in Figure 4B. Figure 4C shows a section of the data in Figure 4B in finer temporal resolution. Note the variance in shape and amplitude between spikes of the same class (identified with a marker of the same gray level) due to the additive background noise. Figure 5 shows another example with noise level 0.15, in which classification is considerably more difficult than in the first data set. Here, the three spike classes share the same peak amplitudes and very similar widths and shapes. The differences

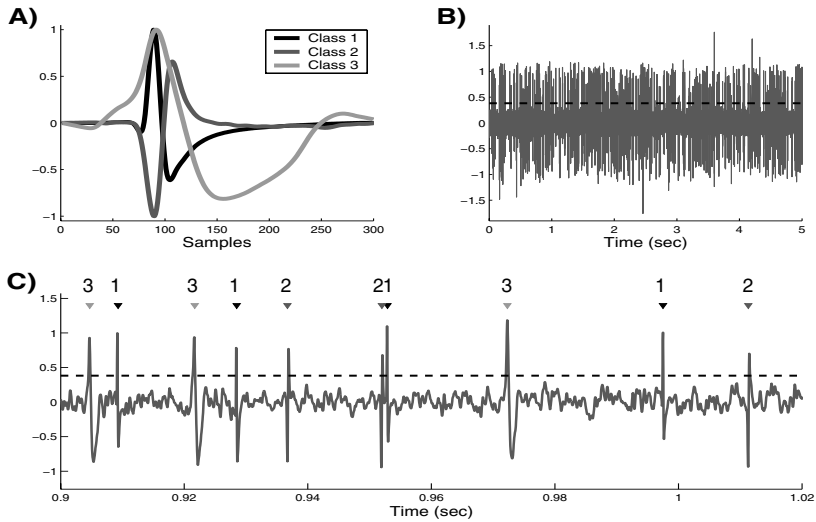

Figure 4: Simulated data set used for spike sorting. (A) The three template spike shapes. (B) The previous spikes embedded in the background noise. (C) The same data with a magnified timescale. Note the variability of spikes from the same class due to the background noise.

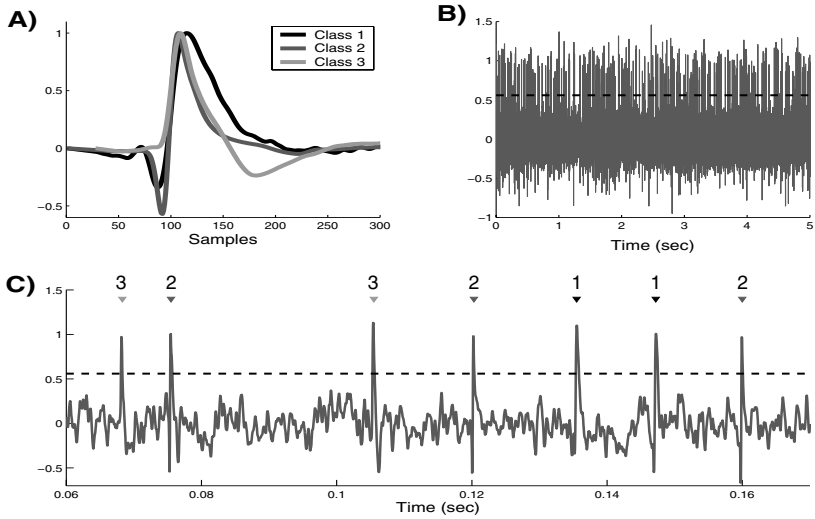

Figure 5: Another simulated data set. (A) The three template spike shapes. (B) The previous spikes embedded in the background noise. (C) The same data with a magnified timescale. Here the spike shapes are more difficult to differentiate. Note in the lower plot that the variability in the spike shapes makes their classification difficult.

between them are relatively small and temporally localized. By adding the background noise, it appears to be very difficult to identify the three spike classes (see Figure 5C). As with the previous data set, the variability of spikes of the same class is apparent.

All the data sets used in this article are available on-line at www.vis.caltech.edu/∼rodri.

5 Results

The method was tested using four generic examples of 60 sec length, each simulated at four different noise levels, as described in the previous section. Since the first example was relatively easy to cluster, in this case we also generated four extra time series with higher noise levels.

5.1 Spike Detection. Figures 4 and 5 show two of the simulated data sets. The horizontal lines drawn in Figures 4B and C and 5B and C are the thresholds for spike detection using equation 3.1. Table 1 summarizes the performance of the detection procedure for all data sets and noise levels. Detection performances for overlapping spikes (i.e., spike pairs within 64 data points) are reported separately (values in brackets). Overlapping

Table 1: Number of Misses and False Positives for the Different Data Sets.

Example Number (Noise Level)	Number of Spikes	Misses	False Positives
Example 1 [0.05]	3514 (785)	17 (193)	711
[0.10]	3522 (769)	2 (177)	57
[0.15]	3477 (784)	145 (215)	14
[0.20]	3474 (796)	714 (275)	10
Example 2 [0.05]	3410 (791)	0 (174)	0
[0.10]	3520 (826)	0 (191)	2
[0.15]	3411 (763)	10 (173)	1
[0.20]	3526 (811)	376 (256)	5
Example 3 [0.05]	3383 (767)	1 (210)	63
[0.10]	3448 (810)	0 (191)	10
[0.15]	3472 (812)	8 (203)	6
[0.20]	3414 (790)	184 (219)	2
Example 4 [0.05]	3364 (829)	0 (182)	1
[0.10]	3462 (720)	0 (152)	5
[0.15]	3440 (809)	3 (186)	4
[0.20]	3493 (777)	262 (228)	2

Notes: Noise level is represented in terms of its standard deviation relative to the peak amplitude of the spikes. All spike classes had a peak value of 1. Values in brackets are for overlapping spikes.

spikes hamper the detection performance because they are detected as single events when they appear too close in time.

In comparison with the other examples, a relatively large number of spikes were not detected in data set 1 for the highest noise levels (0.15 and 0.2). This is due to the spike class with opposite polarity (class 2 in Figure 4). In fact, setting up an additional negative threshold reduced the number of misses from 145 to 5 for noise level 0.15 and from 714 to 178 for 0.2. In the case of the overlapping spikes, this reduction is from 360 to 52 and from 989 to 134, respectively. In all other cases, the number of undetected spikes was relatively low.

With the exception of the first two noise levels in example 1 and the first noise level in example 3, the number of false positives was very small (less than 1%). Lowering the threshold value in equation 3.1 (e.g., 3.5 σ_n) would indeed reduce the number of misses but also increase the number of false positives. The optimal trade-off between number of misses and false positives depends on the experimenter's preference, but we remark that the automatic threshold of equation 3.1 gives an optimal value for different noise levels. In the case of example 1 (noise level 0.05 and 0.1) and example 3 (noise level 0.05), the large number of false positives is exclusively due to double detections. Since the noise level is very low in these cases, the threshold is also low, and consequently, the second positive peak of the class 3 spike shown in Figure 4 is detected. One solution would be to take a higher threshold value (e.g., 4.5 σ_n), but this would not be optimal for high

noise levels. Although double detections decrease the performance of the detection algorithm, it does not represent a problem when considering the whole spike sorting procedure. In practice, the false positives show up as an additional cluster that can be disregarded later. For further testing of the clustering algorithm, the complete data set of simulated spikes (with both the detected and the undetected ones) will be used.

5.2 Feature Extraction. Figure 6 shows the wavelet coefficients for spikes in the data set shown in Figure 4A and Figure 5B. For clarity, wavelet coefficients of overlapping spikes are not plotted. Coefficients corresponding to individual spikes are superimposed, each representing how closely the spike waveform matches the wavelet function at a particular scale and time. Coefficients are organized in detail levels ($D_{1–4}$) and a last approximation (A_4), which correspond to the different frequency bands in which spike shapes are decomposed. Especially in Figure 6A, we observe that some of the coefficients cluster around different values for the different spike classes, thus being well suited for classification. Most of these coefficients are chosen by the KS test, as shown with black markers. For comparison, the 10 coefficients with maximum variance are also marked. It is clear from this figure that coefficients showing the best discrimination are not necessarily the ones with the largest variance. In particular, the maximum variance criterion misses several coefficients from the high-frequency scales (D_1–D_2) that allow a good separation between the different spike shapes.

Figure 7 discloses the distribution of the 10 best wavelet coefficients from Figure 6B (in this case, including coefficients corresponding to overlapping spikes) using the KS criterion versus the maximum variance criterion. Three wavelet coefficients out of the ten selected using the KS criterion show a deviation from normality that is not associated with multimodal distribution: coefficient 42, showing a skewed distribution, and coefficients 19 and 10, which, in addition to skewed distribution, have significant kurtosis mainly due to the outliers introduced by the overlapping spikes. In the remaining cases, the KS criterion selected coefficients with a multimodal distribution. In contrast, with the exception of coefficient 20, the variance criterion selects coefficients with a uniform distribution that hampers classification.

For the same data, in Figure 8 we show the best three-dimensional (3D) projections of the wavelet coefficients selected with the KS criterion (Figure 8A), the variance criterion (Figure 8B) and projections of the first three principal components (Figure 8C). In all cases, the clustering was done automatically with SPC and is represented with different gray levels. We observe that using the KS criterion, it is possible to clearly identify the three clusters. In contrast, when choosing the coefficients with the largest variance, it is possible to identify two out of three clusters, and when using the first three principal components, only a single cluster is detected (the number of classification errors is shown in Table 2, example 2, noise 0.15). Note also that the cluster shapes can be quite elongated, thus challenging any cluster-

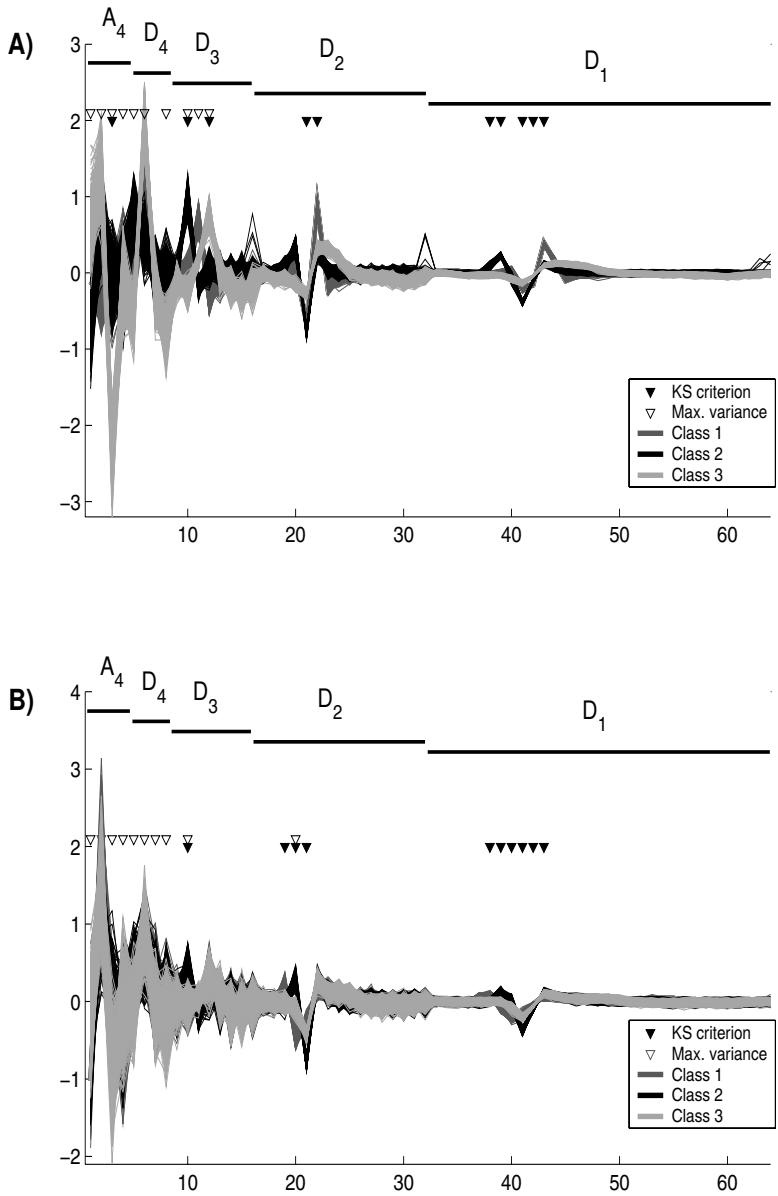

Figure 6: Wavelet transform of the spikes from Figure 4 and Figure 5 (panes A and B, respectively). Each curve represents the wavelet coefficients for a given spike, the gray levels denoting the spike class after clustering with SPC. (A) Several wavelet coefficients are sensitive to localized features. (B) Separation is much more difficult due to the similarity of the spike shapes. The markers show coefficients selected based on the variance criteria and coefficients selected based on deviation from normality. D_1–D_4 are the detail levels, and A_4 corresponds to the last approximation level.

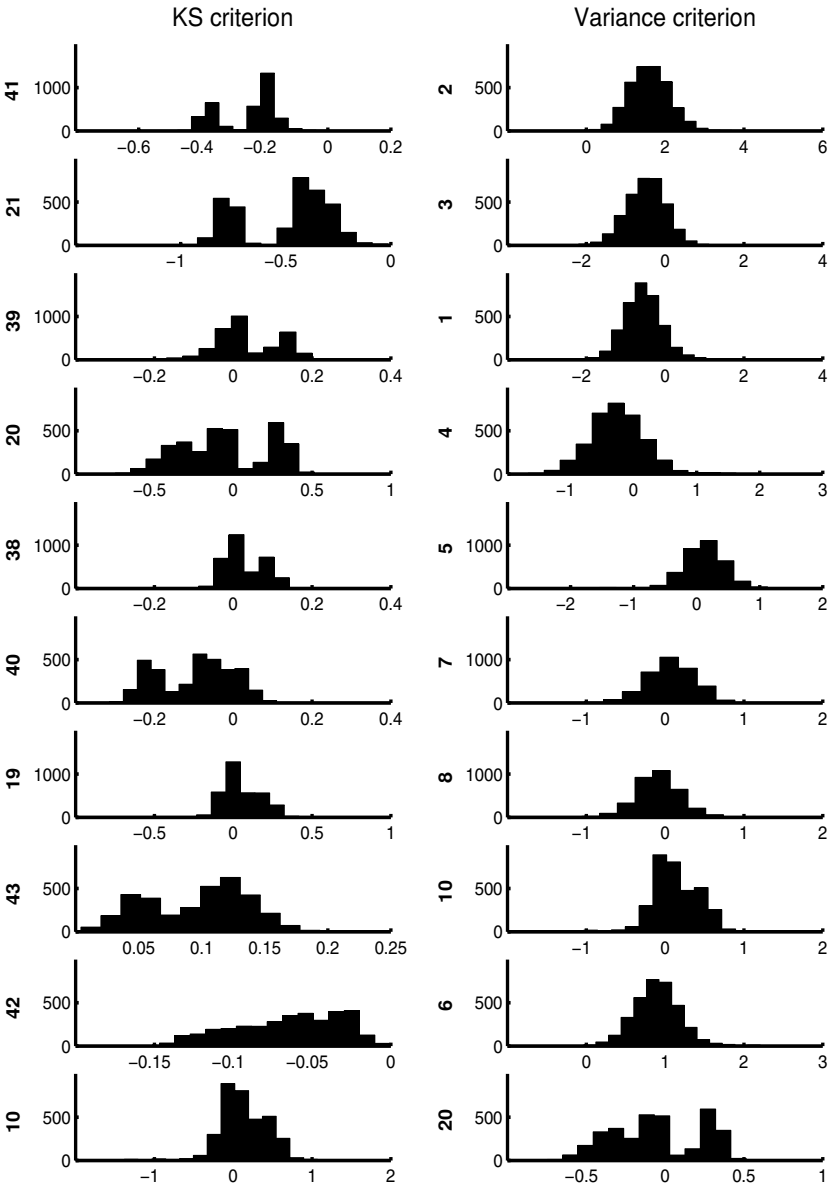

Figure 7: Distribution of the wavelet coefficients corresponding to Figure 6B. (A) The coefficients selected with the Kolmogorov-Smirnov criterion. (B) The coefficients selected with a maximum variance criterion. The coefficient number is at the left of each panel. Note that the first criterion is more appropriate as it selects coefficients with multimodal distributions.

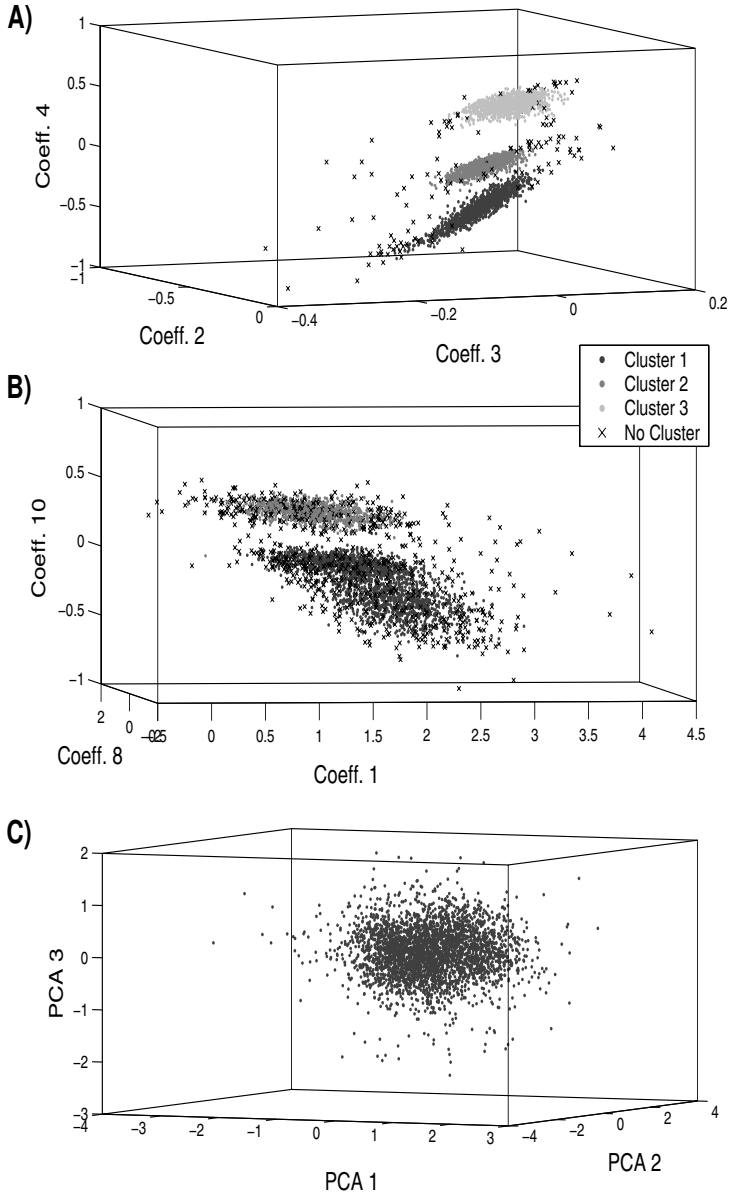

Figure 8: Best projection of the wavelet coefficients selected with the (A) KS criterion and the (B) variance criterion. (C) The projection on the first three principal components. Note that only with the wavelet coefficients selected by using the KS criterion is it possible to separate the three clusters. Clusters assignment (shown with different gray levels) was done after use of SPC.

Table 2: Number of Classification Errors for All Examples and Noise Levels Obtained Using SPC, *K*-Means, and Different Spike Features.

Example Number (Noise Level)	Number of Spikes	SPC				*K*-means	
		Wavelets	PCA	Spike Shape	Feature Set	Wavelets	PCA
Example 1 [0.05]	2729	1	1	0	863	0	0
[0.10]	2753	5	17	0	833	0	0
[0.15]	2693	5	19	0	2015 (2)	0	0
[0.20]	2678	12	130	24	614	17	17
[0.25]	2586	64	911	266	1265 (2)	69	68
[0.30]	2629	276	1913	838	1699 (1)	177	220
[0.35]	2702	483	1926 (2)	1424 (2)	1958 (1)	308	515
[0.40]	2645	741	1738 (1)	1738 (1)	1977 (1)	930	733
Example 2 [0.05]	2619	3	4	2	502	0	0
[0.10]	2694	10	704	59	1893 (1)	2	53
[0.15]	2648	45	1732 (1)	1054 (2)	2199 (1)	31	336
[0.20]	2715	306	1791 (1)	2253 (1)	2199 (1)	154	740
Example 3 [0.05]	2616	0	7	3	619	0	1
[0.10]	2638	41	1781	794	1930 (1)	850	184
[0.15]	2660	81	1748 (1)	2131 (1)	2150 (1)	859	848
[0.20]	2624	651	1711 (1)	2449 (1)	2185 (1)	874	1170
Example 4 [0.05]	2535	1	1310	24	1809 (1)	686	212
[0.10]	2742	8	946 (2)	970 (2)	1987 (1)	271	579
[0.15]	2631	443	1716 (2)	1709 (1)	2259 (1)	546	746
[0.20]	2716	1462 (2)	1732 (1)	1732 (1)	1867 (1)	872	1004
Average	2662	232	1092	873	1641	332	371

Notes: In parentheses are the number of correct clusters detected when different from 3. The numbers corresponding to the example shown on Figure 8 are underlined.

ing procedure based on Euclidean distances to the cluster centers, such as *K*-means.

5.3 Clustering of the Spike Features. In Figure 9, we show the performance of the algorithm for the first data set (shown in Figure 4). In Figure 9A, we plot the cluster sizes as a function of the temperature. At a temperature $T = 0.02$, the transition to the superparamagnetic phase occurs. As the temperature is increased, a transition to the paramagnetic regime takes place at $T = 0.12$. The temperature $T = 0.02$ (vertical dotted line) is determined for clustering based on the criterion described in section 3.3. In Figure 9B, we see the classification after clustering and in Figure 9C the original spike shapes (without the noise). In this case, spike shapes are easy to differentiate due to the large negative phase of the class 1 spikes and the initial negative peak of the class 3 spikes.

Figure 10 shows the other three data sets with a noise level 0.1. In all these cases, the classification errors were very low (see Table 2). Note also that many overlapping spikes were correctly classified (those in color), especially when the latency between the spike peaks was larger than about 0.5 ms. Pairs of spikes appearing with a lower time separation are not clustered

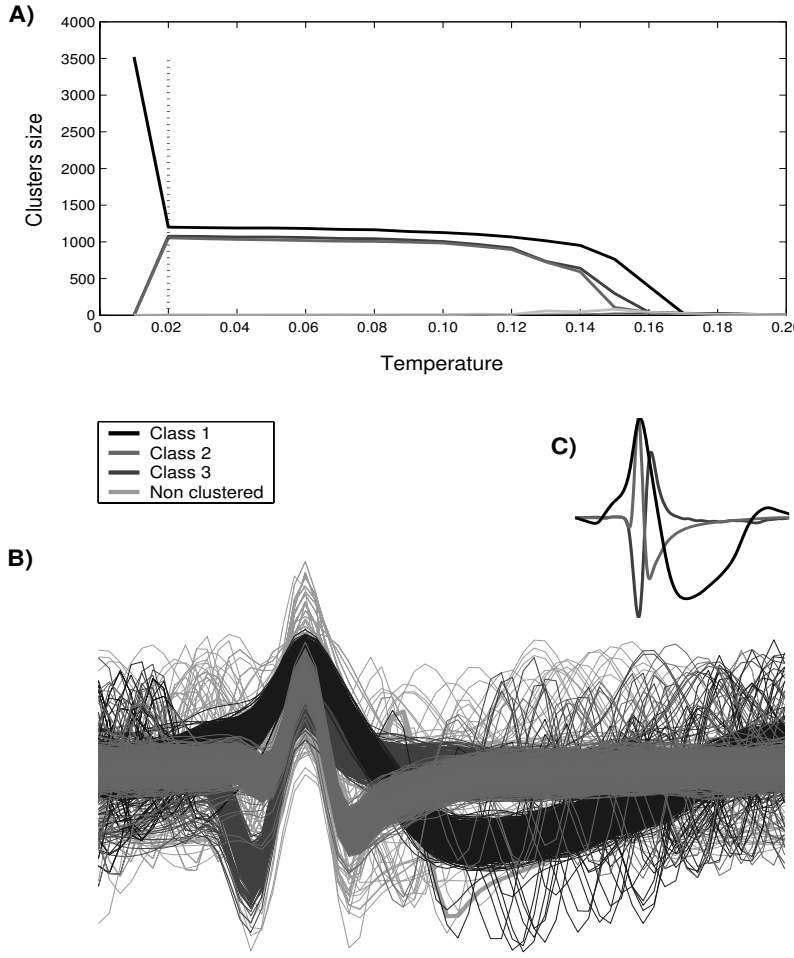

Figure 9: (A) Cluster size vs. temperature. Based on the stability criterion of the clusters, a temperature of 0.02 was automatically chosen for separating the three spike classes. (B) All spikes with gray levels according to the outcome of the clustering algorithm. Note the presence of overlapping spikes. (C) Original spike shapes.

by the algorithm (in gray) but can, in principle, be identified in a second stage by using the clustered spikes as templates. Then one should look for the combination (allowing delays between the spike templates) that best reproduces the nonclustered spike shapes. This procedure is outside the scope of this study and will not be further addressed.

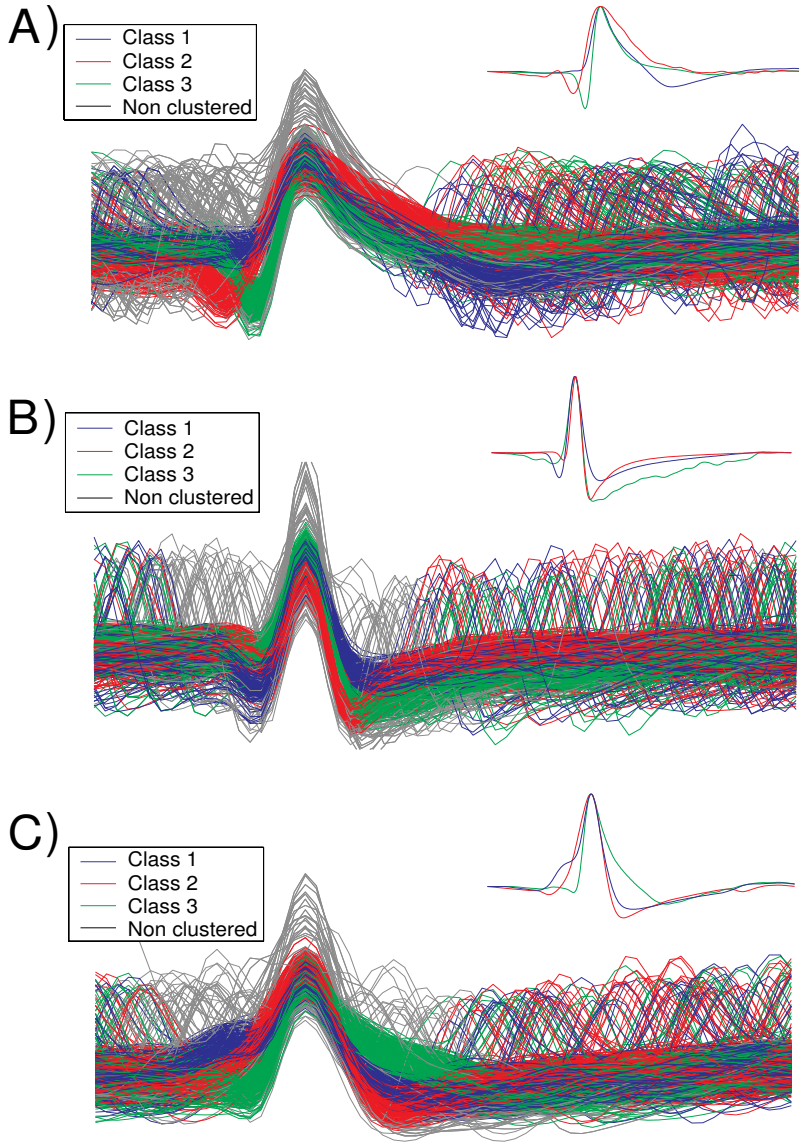

Figure 10: Outcome of the clustering algorithm for the three remaining examples. The inset plots show the original spike shapes. Most of the classification errors (gray traces) are due to overlapping spikes with short temporal separation.

5.4 Comparison with Other Spike Features. Errors of spike identification cumulatively derive from two sources: incorrect feature extraction and incorrect clustering. First, we compared the discrimination power of wavelets at different noise levels with other feature extraction methods using the same clustering algorithm, SPC. Specifically, we compared the outcome of classification with wavelets, principal component analysis (using the first three principal components), the whole spike shape, and a fixed set of spike features. The spike features were the mean square value of the signal (energy), the amplitude of the main peak, and the amplitude of the peaks preceding and following it. The width of the spike (given by the position of the three peaks) was also tested but found to worsen the spike classification. Table 2 summarizes the results (examples 1 and 2 were shown in Figures 4 and 5, respectively). Performance was quantified in terms of the number of classification errors and the number of clusters detected. Values in brackets denote the number of clusters detected, when different from 3. Errors due to overlapping spikes are not included in the table (but overlapping spikes were also inputs to the algorithm). Since example 1 was easier to cluster, in this case we also analyzed four higher noise levels.

In general, the best performance was achieved using the selected wavelet coefficients as spike features. In fact, with wavelets, all three clusters are correctly detected, with the exception of example 4, noise level 0.2. Using the other features, only one cluster is detected when increasing the noise level. Considering all wavelet coefficients as inputs to SPC (i.e., without the selection based on the KS test) gave nearly the same results as those obtained using the entire spike shape (not shown). This is not surprising since the wavelet transform is linear, which implies rotations or rescaling of the original high-dimensional space. The clear benefit is introduced when considering only those coefficients that allow a good separation between the clusters, using the KS test. Note that PCA features gave slightly worse results than those using the entire spike shapes (and clearly worse than using wavelets). Therefore, PCA gives a reasonable solution when using clustering algorithms that cannot handle high-dimensional spaces (although this is not a problem for SPC). The fixed features, such as peak amplitudes and mean squared values, were much less efficient. This was expected since spike classes were generated with the same peak value and in some cases also with very similar shapes that could not be discriminated by these features. We remark that the number of detected clusters decreases monotonically with noise level, but the number of classification errors does not necessarily increase monotonically because even when three clusters are correctly recognized, a large number of spikes may remain unassigned to any of them (see, e.g., example 4 with PCA for noise levels 0.05 and 0.1).

5.5 Comparison with Other Clustering Algorithms. A number of different clustering algorithms can be applied for spike sorting, and the choice of an optimal one is important in order to exploit the discrimination power

of the feature space. The most used algorithms are supervised and usually assume gaussian shapes of the clusters and specific properties of the noise distribution. In order to illustrate the difference with these methods, we will compare results using SPC with those obtained using *K*-means. The partitioning of data by *K*-means keeps objects of the same cluster as close as possible (using Euclidean distance in our case) and as far as possible from objects in the other clusters. The standard *K*-means algorithm leaves no unclassified items, but the total number of clusters should be predefined (therefore being a supervised method). These constraints simplify the clustering problem and give an advantage to *K*-means in comparison to SPC, since in the first case, we know that each object should be assigned to one of the three clusters. The right-most two columns of Table 2 show the clustering performance using K-means with wavelets and PCA. Despite being unsupervised, SPC applied on the wavelet features gives the best performance. For spike shapes relatively easy to differentiate (Table 2, examples 1 and 2), the outcomes with wavelets are similar using K-means or SPC. However, the advantage of SPC with wavelet becomes apparent when spike shapes are more similar (Table 2, examples 3 and 4). We remark that with SPC, points may remain unclassified, whereas with *K*-means, all points are assigned to one of the three clusters (thus having at least a 33% chance of being correctly classified). This led K-means to outperform SPC for example 4, at noise level 0.2, where only two out of three clusters were identified with SPC. In general, the number of classification errors using PCA with K-means is higher than the ones using wavelets with K-means. The few exceptions where PCA outperformed wavelets with using K-means (example 3 at noise level 0.1; example 4 at noise level 0.05) can be attributed to the presence of more elongated cluster shapes obtained with wavelets that *K*-means fails to separate.

5.6 Simulations with Nongaussian Spike Distributions. In this section, we consider conditions of real recordings that may compromise the performance of the proposed clustering algorithm. In particular, we will simulate the effect of an electrode moving with respect to one of the neurons, bursting activity, and a correlation between the spikes and the local field potential. Clearly, these situations are difficult to handle with algorithms that assume a particular distribution of the noise. In fact, all of these cases add a nongaussian component to the spike shape variance.

For simulating a moving electrode, we use the example shown in Figure 5 (example 2, noise 0.15 in Table 2), but in this case, we progressively decreased the amplitude of the first spike class (linearly) with time from a value of 1 at the beginning of the recording to 0.3 at the end. Using SPC with wavelet, it was still possible to detect all three clusters, and from a total of 2692 spikes, the number of classification errors was only of 48.

Second, we simulated a bursting cell based also on the example shown in Figure 5. The first class consisted of three consecutive spikes with ampli-

tudes 1.0, 0.7, and 0.5, separated by 3 ms in average (SD = 1, range, 1–5 ms). From a total of 2360 spikes, again the three clusters were correctly detected, and we had 25 classification errors using wavelets with SPC.

Finally we considered a correlation between the spike amplitudes and the background activity, similar to the condition when spikes co-occur with local field events. We used the same spike shapes and noise level shown in Figure 5, but the spike amplitudes varied from 0.5 to 1 depending on the amplitude of the background activity at the time of the spike (0.5 when the background activity reached its minimum and 1.0 when it reached its maximum). In this case, again, the three clusters were correctly detected, and we had 439 classification errors from a total of 2706 spikes.

6 Discussion

We presented a method for detection and sorting neuronal multiunit activity. The procedure is fully unsupervised and fast, thus being particularly interesting for the classification of spikes from a large number of channels recorded simultaneously. To obtain a quantitative measure of its performance, the method was tested on simulated data sets with different noise levels and similar spike shapes. The noise was generated by superposition of a large number of small-amplitude spikes, resembling characteristics of real recordings. This makes the spectral characteristics of noise and spikes similar, thus increasing the difficulty in detection and clustering. The proposed method had an overall better performance than conventional approaches, such as using PCA for extracting spike features or *K*-means for clustering.

Spike detection was achieved by using an amplitude threshold on the high-pass filtered data. The threshold value was calculated automatically using the median of the absolute value of the signal. The advantage of this estimation, rather than using the variance of the overall signal, is that it diminishes the dependence of the threshold on the firing rate and the peak-to-peak amplitude of the spikes, thus giving an improved estimation of the background noise level. Indeed, high firing rates and high spike amplitudes lead to an overestimation of the appropriate threshold value. In terms of the number of misses and number of false positives, the proposed automatic detection procedure had good performance for the different examples and noise levels.

The advantage of using the wavelet transform as a feature extractor is that very localized shape differences of the different units can be discerned. The information about the shape of the spikes is distributed in several wavelet coefficients, whereas with PCA, most of the information about the spike shapes is captured only by the first three principal components (Letelier & Weber, 2000; Hulata, Segev, Shapira, Benveniste, & Ben-Jacob, 2000; Hulata, Segev, & Ben-Jacob, 2002), which are not necessarily optimal for cluster identification (see Figure 7). Moreover, wavelet coefficients are localized in time. In agreement with these considerations, a better performance of

wavelet coefficients in comparison with PCA was shown for several examples generated with different noise levels. For comparison, we also used the whole spike shape as input to the clustering algorithm. As shown in Table 2, the dimensionality reduction achieved with the KS test clearly improves the clustering performance. Since wavelets are a linear transform, using all the wavelet coefficients yields nearly the same results as taking the entire spike shape (as it is just a rescaling of the space). Since the need of a low-dimensional space is a limiting factor for many clustering algorithms, the dimensionality reduction achieved by combining wavelets with the KS test may have a broad range of interest.

The use of wavelets for spike sorting has been proposed recently by Letelier et al. (2000) and Hulata et al. (2000, 2002). Our approach differs from theirs in several aspects, most notably by the implementation of our algorithm as a single unsupervised process. One key feature of our algorithm is the choice of wavelet coefficients by using a Kolmogorov-Smirnov test of normality, thus selecting features that give an optimal separation between the different clusters. Letelier et al. (2000) suggested to visually select those wavelet coefficients with the largest mean, variance, and, most fundamental, multimodal distribution. However, neither a large average nor a large variance entitles the given coefficient to be the best separator. In contrast, we considered only the multimodality of the distributions. In this respect, we showed that coefficients in the low-frequency bands, with a large variance and uniform distribution, are not appropriate for the separation of distinct clusters. In contrast, they introduce dimensions with nonsegregated distributions that in practice may compromise the performance of the clustering algorithm. A caveat of the KS test as an estimator of multimodality is that it can also select unimodal nongaussian distributions (those that are skewed or have large kurtosis). In fact, this was the case of three coefficients shown in Figure 7. Despite this limitation, the selection of wavelet coefficients with the KS test gave optimal results that indeed outperformed other feature selection methods.

The main caveat of PCA is that eigenvectors accounting for the largest variance of the data are selected, but these directions do not necessarily provide the best separation of the spike classes. In other words, it may well be that the information for separating the clusters is represented in principal components with low eigenvalues, which are usually disregarded. In this respect, our method is more reminiscent of independent component analysis (ICA), where directions with a minimum of mutual information are chosen. Moreover, it has been shown that minimum mutual information is related to maximum deviation from normality (Hyvarinen & Oja, 2000). Hulata et al. (2000, 2002) proposed a criterion based on the mutual information of all pairs of cluster combinations for selecting the best wavelet packet coefficients. However, such an approach cannot be implemented in an unsupervised way. In fact, Hulata and coworkers used previous knowledge of the spike classes for selecting the best wavelet packets. A second caveat

is the difficulty of estimating mutual information (Quian Quiroga, Kraskov, Kreuz, & Grassberger, 2002) in comparison with the KS test of normality.

The second stage of the clustering algorithm is based on the use of superparamagnetic clustering. This method is based on *K*-nearest neighbor interactions and therefore does not require low variance, nonoverlapping clusters, or a priori assumptions of a certain distribution of the data (e.g., gaussian). Superparamagnetic clustering has already been applied with excellent results to several clustering problems (Blatt et al., 1996, 1997; Domany, 1999). In our study, we demonstrated for the first time its application to spike sorting. Moreover, it is possible to automatically locate the superparamagnetic regime, thus making the entire sorting procedure unsupervised. Besides the obvious advantage of unsupervised clustering, we compared the results obtained with SPC to those obtained using *K*-means (with Euclidean distance). Although this comparison should not be generalized to all existing clustering methods, it exemplifies the advantages of SPC over methods that rely on the presence of gaussian distributions, clusters with centers inside the cluster (see Figure 1 for counterexamples), nonoverlapping clusters with low variance, and others. The performance of *K*-means could in principle be improved by using another distance metric. However, this would generally imply assumptions about the noise distribution and its interference with spike variability. Such assumptions may improve the clustering performance in some situations but may also be violated in other conditions of real recordings. Note that this comparison was in principle unfair to SPC since *K*-means is a supervised algorithm where the total number of clusters is given as an extra input. Of course, the total number of clusters is usually not known in real recording situations. In general, besides the advantage of being unsupervised, SPC showed better results than the ones obtained with *K*-means.

Overall, the presented results show an optimal performance of the clustering algorithm in situations that resemble real recording conditions. However, we should stress that this clustering method should not be taken as a black box giving the optimal spike sorting. When possible, it is always desirable to confirm the validity of the results based on the shape and variance of the spike shapes, the interspike interval distribution, the presence of a refractory period, and so forth. Finally, we anticipate the generalization of the method to tetrode recordings. Indeed, adding spike features from adjacent channels should improve spike classification and reduce ambiguity.

Acknowledgments

We are very thankful to Richard Andersen and Christof Koch for support and advice. We also acknowledge very useful discussions with Noam Shental, Moshe Abeles, Ofer Mazor, Bijan Pesaran, and Gabriel Kreiman. We are in debt to Eytan Domany for providing us the SPC code and to Alon Nevet

who provided the original spike data for the simulation. This work was supported by the Sloan-Swartz foundation and DARPA.

References

Abeles, M., & Goldstein, M. (1977). Multispike train analysis. *Proc. IEEE, 65*, 762–773.

Binder, K., & Heermann, D. W. (1988). *Monte Carlo simulations in statistical physics: An introduction*. Berlin: Springer-Verlag.

Blatt, M., Wiseman, S., & Domany, E. (1996). Super-paramagnetic clustering of data. *Phys. Rev. Lett., 76*, 3251–3254.

Blatt, M., Wiseman, S., & Domany, E. (1997). Data clustering using a model granular magnet. *Neural Computation, 9*, 1805–1842.

Chui, C. (1992). *An introduction to wavelets*. San Diego, CA: Academic Press.

Domany, E. (1999). Super-paramagnetic clustering of data: The definitive solution of an ill-posed problem. *Physica A, 263*, 158–169.

Donoho, D., & Johnstone, I. M. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika, 81*, 425–455.

Fee, M. S., Mitra, P. P., & Kleinfeld, D. (1996). Variability of extracellular spike waveforms of cortical neurons. *J. Neurophysiol., 76*, 3823–3833.

Harris, K. D., Henze, D. A., Csicsvari, J., Hirase, H., & Buzsáki, G. (2000). Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements. *J. Neurophysiol., 84*, 401–414.

Hulata, E., Segev, R., & Ben-Jacob, E. (2002). A method for spike sorting and detection based on wavelet packets and Shannonís mutual information. *J. Neurosci. Methods, 117*, 1–12.

Hulata, E., Segev, R., Shapira, Y., Benveniste, M., & Ben-Jacob, E. (2000). Detections and sorting of neural spikes using wavelet packets. *Phys. Rev. Lett., 85*, 4637–4640.

Hyvarinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks, 13*, 411–430.

Letelier, J. C., & Weber, P. P. (2000). Spike sorting based on discrete wavelet transform coefficients. *J. Neurosci. Methods, 101*, 93–106.

Lewicki, M. (1998). A review of methods for spike sorting: The detection and classification of neural action potentials. *Network: Comput. Neural Syst., 9*, R53–R78.

Mallat, S. (1989). A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Analysis and Machine Intell., 2*, 674–693.

Pouzat, C., Mazor, O., & Laurent, G. (2002). Using noise signature to optimize spike-sorting and to assess neuronal classification quality. *J. Neurosci. Methods, 122*, 43–57.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). *Numerical recipes in C*. Cambridge: Cambridge University Press.

Quian Quiroga, R., & Garcia, H. (2003). Single-trial event-related potentials with wavelet denoising. *Clin. Neurophysiol., 114*, 376–390.

Quian Quiroga, R., Kraskov, A., Kreuz, T., & Grassberger, P. (2002). Performance of different synchronization measures in real data: A case study on electroencephalographic signals. *Phys. Rev. E, 65*, 041903.

Quian Quiroga, R., Sakowicz, O., Basar, E., & Schürmann, M. (2001). Wavelet Transform in the analysis of the frequency composition of evoked potentials. *Brain Research Protocols, 8*, 16–24.

Samar, V. J., Swartz, K. P., & Raghveer, M. R. (1995). Multiresolution analysis of event-related potentials by wavelet decomposition. *Brain and Cognition, 27*, 398–438.

Wolf, U. (1989). Comparison between cluster Monte Carlo algorithms in the Ising spin model. *Phys. Lett. B, 228*, 379–382.

Received December 3, 2002; accepted January 30, 2004.