32nd Annual International Conference of the IEEE EMBS
Buenos Aires, Argentina, August 31 - September 4, 2010

Spike Detection in Extracellular Recordings by Hybrid Blind Beamforming

Michal Natora, Felix Franke and Klaus Obermayer

Abstract— **In the case of extracellular recordings, spike detection algorithms are necessary in order to retrieve information about neuronal activity form the data. We present a new spike detection algorithm which is based on methods from the field of blind equalization and beamforming. In contrast to existing approaches, our method estimates several waveforms directly from the data and corresponding linear filters are constructed. The estimation is done in an unsupervised manner, and the few parameters of the algorithm are intuitive to set. The algorithm allows for superior detection performance, even when multiple neurons with various waveforms are present in the data. We compare our method with current state-of-the-art spike detection algorithms, and show that the proposed method achieves favorable results.**

I. Introduction

Extracellular recordings with electrodes constitute one of the main techniques for acquiring data from the central nervous system in order to study the neuronal code. Information in this system is transmitted by short electric impulses, called action potentials or, hereinafter, spikes. One of the first processing stages of the recorded data, hence, consist of identifying the occurrence times of these spikes. To this end, various spike detection algorithms have been developed. To give a structured overview of the recent development in this field, we use a categorization scheme based on the working principle of the methods. Note, that although the spike detection stage is one of the earliest, basically all algorithm require already some pre-processing. This includes a band pass filtering (usually between 0.5kHz and 10kHz), and a zero mean normalization. In the following, we will still refer to this kind of pre-processed data as "raw" data, since all techniques rely on this initial step.

The first category of spike detection methods assumes that spikes exhibit a larger amplitude than noise fluctuations. Hence, spikes can be detected as data segments whose amplitude cross a certain threshold value. In [1] three different variations of this detection paradigm were described, including maximum, minimum and absolute value thresholding.

The principle of the second category is based on the transient nature of a spike, thus, spikes can be detected by measuring some quantity describing the discontinuity of data. An example is the nonlinear energy operator which takes into account instantaneous energy and frequency, and which was used for spike detection in [2]. Further adaptations of this method to neural data have been proposed in [3].

The algorithms falling into the third category rely on the fact that spikes from a specific neuron exhibit a characteristic waveform. The similarity between a data segment and a specified waveform decides whether the considered data segment contains a spike. When the actual waveform in the data is unknown, a generic approach can be used. For example in [4], [5], biorthogonal respectively coiflet mother wavelets were used, since they exhibit a certain similarity in shape to waveforms found in some real recordings, and a spike is said to be detected when a specific function of wavelet coefficients exceeds a threshold value. In contrast, unsupervised estimation (also called blind estimation) of the waveform has been performed in [6] by linear prediction, in [7] by automatic threshold setting, or in [8] by using the cepstrum of bispectrum. The extracted waveform template was then used for constructing a linear filter, on whose output spikes were detected by thresholding.

The choice which algorithm should be used in an application surely depends on the two important aspects of computational complexity and detection performance. Limited power and computing recourses, as encountered in implantable circuits, restrict applicable algorithm to have a very low computational load, hence mostly methods from the first and few from the second category are used [9]. When not limited by such constraints, it is favorable with respect to the detection performance to use algorithms belonging to the third category. This is motivated by the fact that, given the waveform and the noise covariance matrix, the matched filter or equivalently the minimum variance distortionless response beamformer (MVDR), is the optimal detector in the case of Gaussian noise [10].

The aforementioned methods dedicated to blind estimation suffer from a main drawback, namely, they construct only a single filter. In many experimental situations, however, spikes from more than one neuron, having distinct waveforms, are present in the electrode recordings. The single filter either captures just one waveform, meaning that spikes from the other neurons will be detected poorly, or the filter is an average filter which will have a sub-optimal response to spikes from all the neurons. This problem aggravates the more neurons are present, and the more the waveforms are distinct, which is especially the case in multi channel recording devices, such as tetrodes [11].

In this contribution, we propose a new spike detection algorithm which overcomes this drawback. The algorithm was derived by considering the spike detection task as a

M. Natora and K. Obermayer are with the Institute for Software Engineering and Theoretical Computer Science, Berlin Institute of Technology, 10623 Berlin, Germany. Email: natora@cs.tu-berlin.de

F. Franke and K. Obermayer are with the Bernstein Center for Computational Neuroscience, 10115 Berlin, Germany

978-1-4244-4124-2/10/$25.00 ©2010 IEEE

blind equalization problem in a multiple-input, single-output system. The algorithm consists of a two step procedure: An iterative algorithm based on higher order statistics and deflation is used in the first step, which leads to an initial filter estimate. In the next step, the minimum variance distortionless response beamformers are calculated leading to an increased detection performance (see Fig. 1 for a graphical representation of the whole algorithm). Because we use techniques from both fields, i.e. blind equalization and classical beamforming, in the context of spike detection, we call our method hybrid blind beamforming for spike detection (*HBBSD*).

The rest of the paper is organized as follows: In Sec. II the algorithm and all its individual steps are described. The evaluation of its performance and comparison with existing spike detection methods are presented in Sec. III. Conclusive remarks are given in Sec. IV.

II. Methods

A. Model of recorded data

In order to derive a well motivated algorithm avoiding heuristics as much as possible, the recorded data has to be described by some signal model. In the neuroscience community, it is widely accepted that the data x recorded at an electrode can often be represented as an linear sum of convolutions of the intrinsic spike trains s_i with constant waveforms $\boldsymbol{q}_i$ and colored Gaussian noise n (having a noise covariance matrix $\boldsymbol{C}$), see e.g. [12], [13]. Explicitly, it is

$$x(t) = \sum_{i=1}^{M} \sum_{\tau} q_i(\tau) s_i(t-\tau) + n(t), \tag{1}$$

where M is the number of neurons whose spikes are present in the recordings. For the sake of clarity, we restricted the model to single channel recordings, i.e. electrodes, but an extension to multichannel data as provided by tetrodes is straightforward.

Since the goal of spike detection is to recover the spike trains s_i from a linear time-invariant system without a priori knowledge about the shape of the waveforms $\boldsymbol{q}_i$, this can be viewed as a blind equalization[1] problem. An overview about this topic and a survey of available methods dealing with such problems can be found in [14].

Most often, M, the number of sources, will be larger than the number of recording channels. In the model of a single electrode as described in Eq. 1, the number of recording channels is equal to one, in which case the generative system is referred to as multiple-input, single-output. In general, it is not possible to extract more sources than available recording channels [14]. In the following, we make explicit use of the unique properties of neural data, such as sparseness and binary alphabet, to overcome this restriction partially.

B. Application of the super-exponential algorithm

The super-exponential algorithm (SEA) developed in [15] achieves blind equalization via filter calculation by higher order cross cumulants. For real valued data, the filter $\boldsymbol{h}$ at iteration $k+1$ is computed as

$$\boldsymbol{h}^{(k+1)} = \frac{\boldsymbol{R}^{-1} \cdot \boldsymbol{d}^{(k)}}{\sqrt{\boldsymbol{d}^{(k)\top} \boldsymbol{R}^{-1} \cdot \boldsymbol{d}^{(k)}}} \tag{2}$$

where $\boldsymbol{R}$ is the data covariance matrix, $(\boldsymbol{R})_{i,j} = \mathrm{cov}(x(t-i), x(t-j))$, and $d^{(k)}(n) = \mathrm{cum}(y^{(k)}(t) : p, x(t-n) : 1)$ denotes the cross-cumulant between p-times $y^{(k)}(t)$ and $x(t-n)$, and $y^{(k)}(t) = \sum_{\tau} h^{(k)}(\tau) x(t-\tau)$ being the filter output.

The algorithm works when the signals s_i are non-Gaussian and when the $\boldsymbol{q}_i$ are stable[2]. In the context of neural recordings, both requirements are surely met. Firstly, the s_i represent the intrinsic spike trains, thus taking values of either 0 or 1, and whose probability density function follow most likely a sparse Bernoulli or Poisson distribution. Secondly, the waveforms $\boldsymbol{q}_i$ are finite impulse response filters, and hence are stable. The SEA algorithm is said to have reached convergence when the difference between two consecutive iterations is small enough (see also Sec. III-C). For convenience, we call the filter obtained at the last iteration simply $\boldsymbol{h}$, instead of $\boldsymbol{h}^{(k_{last})}$.

The choice of the SEA instead of other blind equalization algorithms was motivated by several of its features. It is shown that in the noise-free case, the algorithm converges independently of the initial condition to the globally optimal solution with a super-exponential convergence rate [15]. Since it uses higher order statistics, this property should also hold approximatively when Gaussian noise is present, as higher order cumulants are zero for Gaussian signals. Moreover, the algorithm is not gradient based like Bussgang type algorithms, thus no step size selection is required, which reduces the amount of parameter settings for the user.

For neural data, we chose the order of the cumulant to be $p = 2$. In this case, the vector $\boldsymbol{d}$ is proportional to the skewness, a statistic which is well suited for asymmetric signals, such as the s_i [16].

C. Mode detection in the filter output

The SEA computes a single filter on the basis of a vector $\boldsymbol{d}$ which contains the statistics of all M waveforms. Nevertheless, as it is most likely that the characteristics of the neurons will be different with respect to signal-to-noise ratio, spiking frequency, or shape of waveform, it is expected that the filter will have various responses to the different neuronal waveforms. The idea is to identify spikes which belong to a single component and re-calculate the filter using only these spikes. The identification is done by a technique called mode finding [17]. Firstly, only the maximum values, denoted as m_i, of the filter output y within a certain range L_s are extracted. Then, the probability density function p_m

[1] Often also called blind deconvolution, blind identification, or convolutive blind source separation

[2] Stable in the sense of robust against noise, not in the sense of stationary in time

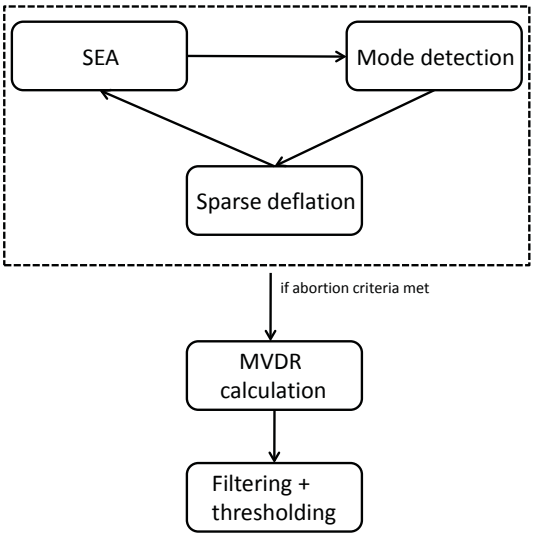

Fig. 1. Schematic illustration of the proposed algorithm HBBSD. The algorithm starts with the super-exponential algorithm (SEA), and iterates between SEA, Mode detection and Sparse deflation repetitively, until certain abortion criteria described in Sec. II-E are met. Finally, the MVDR filters are calculated and spike detection is done by thresholding the filter output.

of the m_i is estimated by a kernel density estimator, which in the assumed case of Gaussian noise is favorable to be a Gaussian kernel. The kernel bandwidth is chosen optimally depending on the amount of data [18]. The function p_m will exhibit a high amplitude mode due to noise, and possibly several low amplitude modes caused by spikes [3]. Hence, the second largest mode b_2 is the prominent spike mode, i.e. caused by spikes to which the filter responded the most, and which consequently should be extracted from the data first. All m_i which have a smaller distance to b_2 than to any other mode are considered to belong to b_2. However, modes which are in the range of $\pm 2\sigma_{n_h}$ around b_2 are not regarded as separate modes, whereas σ_{n_h} denotes the standard deviation of noise in the filter output (of filter $\boldsymbol{h}$).This is motivated by the fact that two Gaussian distributions with identical standard deviation do not exhibit two separate modes, unless their means are at least $2\sigma_{n_h}$ apart. This merging of modes is necessary in order to minimize the number of spurious modes which do not represent an individual component but are mere artifacts caused by the kernel smoothing. The number of modes will depend on the number of present neurons and how distinct their properties are, while their amplitude will mostly depend on the firing frequency. Note that, even when two modes are merged which would in fact represent two components, the proposed algorithm should not perform worse than a single filter obtained by SEA.

1) Estimation of the filter output noise variance: To estimate σ_{n_h}, first the mean μ_{n_h} of the filter output noise is estimated. If one can assume that n is zero mean, then this step can be avoided, since then it immediately follows that $\mu_{n_h} = 0$ as well. Otherwise, the probability density function of y is estimated by a Gaussian kernel density estimator as described in the previous section. Making again use of the sparseness of the data, the mean μ_{n_h} is found as the global maximum of this probability density function. As we expect that the response of filter $\boldsymbol{h}$ to spikes is larger than μ_{n_h}, we ignore all values of y which are above μ_{n_h}, since they are likely to contain spikes. Hence, σ_{n_h} is solely estimated on values of y which are smaller than μ_{n_h}.

[3]Due to the large amount of noise samples, the kernel bandwidth will be relatively small, which guarantees that the modes caused by spikes will not be smoothed away.

D. Sparse deflation

In classical algorithms designed for multiple-input, multiple-output systems, sources are extracted one by one using a technique called deflation [19]. As such, one single waveform $\boldsymbol{q}_j$ is estimated via second order statistics, the source s_j is estimated via the convolution of the corresponding filter $\boldsymbol{h}_j$ with x, and the convolution between $\boldsymbol{q}_j$ and s_j is subtracted from the data x. This classical deflation procedure was developed by assuming that the sources are continuous signals, and that the waveforms have to be known only up to a scalar factor. In contrast, the signals representing the occurrences of spikes are discrete and sparse, and, as will be shown in Sec. II-F, the waveforms need to be known without ambiguity. Therefore, we propose an adapted deflation procedure which we call sparse deflation, as it relies on the sparseness of the data. At iteration j data segments $\boldsymbol{x}^j{}_i$ of length L_f are cut out of x around the maxima m_i, $i = 1, ..., K$, which belong to the mode b_2. Finally, the waveform is estimated as the median of all data segments, i.e.

$$\hat{q}_j(t) = \text{med}\left\{x_1^j(t), \ldots, x_K^j(t)\right\} \qquad t = 1, ..., L_f, \qquad (3)$$

where K is the total number of maxima m_i belonging to mode b_2. Instead of subtracting the estimated contribution of source s_j, the data segments $\boldsymbol{x}^j{}_i$ are simply removed from the data. The reduced data set $x \setminus \boldsymbol{x}^j{}_i$, $i = 1, ..., K$, is now used as the starting point for the next iteration of the algorithm. In particular, the steps described in Sec. II-B - II-D are repeated on the updated data $x \setminus \boldsymbol{x}^j{}_{i=1,\ldots,K} =: x$.

E. Abortion criteria

The iteration loop is terminated if at least one of the following criteria is met:

- No spike mode can be identified in the filter output anymore, or the number of spikes belonging to the spike bump is below a relative threshold $minf$
- A maximum number of iteration is reached

If the loop abortion happens after the first iteration already, the filter obtained by Eq. 2 is used for further spike detection instead of the MVDR beamformers.

F. Calculation of the MVDR beamformers

Once the iteration loop described in the previous sections is completed, the final filters used for spike detection are

calculated. Namely, we use the MVDR beamformer which is given by [10]

$$\boldsymbol{f}_i = \frac{\hat{\boldsymbol{C}}^{-1} \cdot \hat{\boldsymbol{q}}_i}{\hat{\boldsymbol{q}}_i^{\top} \cdot \hat{\boldsymbol{C}}^{-1} \cdot \hat{\boldsymbol{q}}_i}, \tag{4}$$

where $\hat{\boldsymbol{C}}$ is the estimate of the noise covariance matrix. The estimate of $\boldsymbol{C}$ is done after the last algorithm iteration, as the deflated data set $x \setminus \boldsymbol{x}^{j=1,\ldots,J}{}_{i=1,\ldots,K}$ contains far less spikes than the original data x allowing for a more accurate noise estimation. After calculating the MVDR beamformers, the data are filtered with each of them, and a spike is declared as detected when the filter output exceeds a certain threshold.

G. Implementation

The proposed algorithm was implemented in MATLAB® version 7.6, but not optimized for maximum computational speed yet. The code and a sample file can be downloaded from the website http://user.cs.tu-berlin.de/∼natora/

III. Performance Evaluation

A. Generation of artificial data

Artificial data were generated according to the model in Eq. 1. The waveforms were constructed from sorted spikes obtained from acute recordings in the prefrontal cortex of macaque monkeys and had a length of about 0.9ms, see Fig. 2. Detailed information about the sorting method and the experimental setup were described in [20]. The spike arrival times were simulated as independent homogeneous Poisson processes with an enforced refractory period of 2ms. The noiseless data were simulated at a sampling frequency of 40kHz and then down-sampled to 10kHz, in order to include the phenomenon of sampling jitter as encountered in real recordings. Gaussian noise with an autocorrelation structure measured in real recordings was simulated by an ARMA process and added to the spike trains (see [20] for more details). All simulated datasets had a length of 6s.

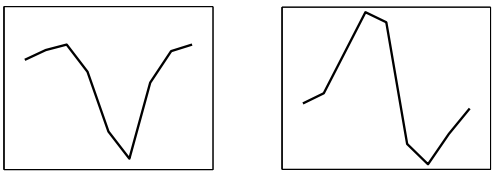

Fig. 2. Waveform templates obtained from extracellular recordings in macaques and used for generation of artificial datasets.

B. Performance assessment

To allow for a better comparison, the same definition of signal-to-noise ratio (SNR) was used as, for example, in [5]. Namely, the SNR of the i-th spike train is defined as the ratio between the norm of the corresponding waveform and the standard deviation of noise,

$$\text{SNR}_i = \frac{||\boldsymbol{q}_i||_\infty}{\sigma_n}. \tag{5}$$

The detection performance of an algorithm was investigated by means of receiver operator characteristic (ROC) curves and the corresponding areas under the curves (AUC), similarly defined as in [7]. The ROC curves were calculated by varying systematically the detection threshold[4] and evaluating the relative number of true positive (TP) and false positive detections (FP), given by

$$\text{TP} = \frac{\text{\# of correct detections}}{\text{\# of inserted spikes}}, \tag{6}$$

$$\text{FP} = \frac{\text{\# of false detections}}{\text{maximal \# of possible false detections}}.$$

A detection was classified as correct, if the detectors response was within ± 0.4ms of the true spiking time. Multiple detections within this time frame were ignored. Consequently, there is a maximum number of possible false detections a detector can produce in a dataset of finite length. By the definition in Eq. 6, both quantities TP and FP are bounded on the interval $[0, 1]$.

C. Parameter settings of considered algorithms

In all subsequent simulations the following parameters were used in the *HBBSD* algorithm: The SEA algorithm was said to have reached convergence if $||\boldsymbol{h}^{(k+1)} - \boldsymbol{h}^{(k)}||_2 \leq 10^{-10}$. The minimal firing frequency $minf$ was set to 2Hz, the filter length was equal to 9 samples ($L_f = L_s = 9$), and a maximum number of 3 filters was allowed. Here we would like to point out that, unlike in some other methods, where the parameters are algorithm specific and thus their value setting is not an obvious task, the parameters of *HBBSP* are biologically motivated, allowing for a reasonable choice of their values. For example, since single channel data is analyzed, it is sound to assume that action potentials from not more than 3 to 4 nearby neurons will be recorded, justifying a maximum filter amount of 3. The filter length can be chosen as the length of a spike, which is most often in the range of 0.4 to 1.0ms [5]. Besides, there exist methods to estimate the filter length even when no biologically motivated a priori knowledge is available. In Fig. 3 it is shown that the filter length L_f has only limited influence on the detection performance of the SEA algorithm. Finally, it is unlikely that neurons in a task relevant brain region will exhibit very low firing frequencies, but, as a matter of fact, the parameter $minf$ could be dropped entirely from the algorithm structure. The algorithms chosen for comparison covered all three categories mentioned in Sec. I. The focus, however, was on methods which make use of waveform information, since in general they achieve the best performance. In the case of amplitude crossing, the absolute value thresholding method was considered, hereinafter abbreviated as *ABS*. The non linear energy operator with a 5 point Bartlett window smoothing (*SNEO*) as described in [2] was chosen, representing a commonly used method based on the transient property of spikes. At last, 3 different methods relying on waveform information were compared.

[4] when several MVDR filters are calculated, the threshold is varied simultaneously for all of them.

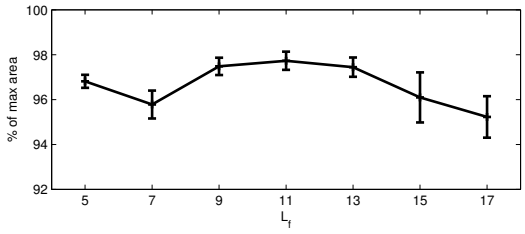

Fig. 3. Same data and evaluation was used as in Sec. III-D. The filter length L_f was varied and the AUC for the SEA algorithm was computed. The bars indicate the standard deviation over ten independent simulations/datasets.

These included the wavelet method (*Wav*) presented in [5], the cepstrum of bispectrum method (*CoB*) from [8], and the classical, single iteration, super-exponential method (*SEA*). The parameters for *Wav* and *CoB* were chosen according to their reference and adapted to the herein considered sampling frequency and spike length.

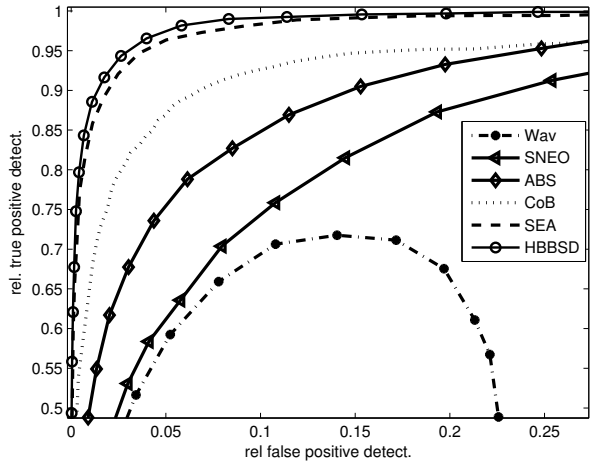

Fig. 4. ROC curves for various spike detection methods. The shown results are an average over 10 independent simulations. Each dataset contained spikes from a single neuron, the signal-to-noise ratio being SNR = 3.0.

D. Performance on data with a single neuron

The first data set contained spikes form a single neuron spiking at a frequency of 25Hz, whereas the waveform is shown on the left in Fig. 2. The ROC curves for every considered method are shown in Fig. 4.

In general, methods which estimate the waveform from the data outperform the generic approaches such as *ABS* and *SNEO*. *Wav* relies on an accordingly chosen mother wavelet by the user. However, if the shape information is not available, and thus the default mother wavelet is used, the performance of this method might be very poor, as indicated in Fig. 4. The decreasing number of true positive detections despite a decreasing threshold is explained by the fact, that *Wav* merges detected spike epochs if they are too close[5].

[5]of course *Wav* will achieve good results and outperform *ABS* and *SNEO* if the waveform is more similar to the used mother wavelet.

The methods which estimate the filter from the data itself show good performance, whereas *HBBSD* achieves the highest score, followed by *SEA* and *CoB*. Based on these findings, we will focus on the comparison of *CoB*, *SEA* and *HBBSD* in the remaining section.

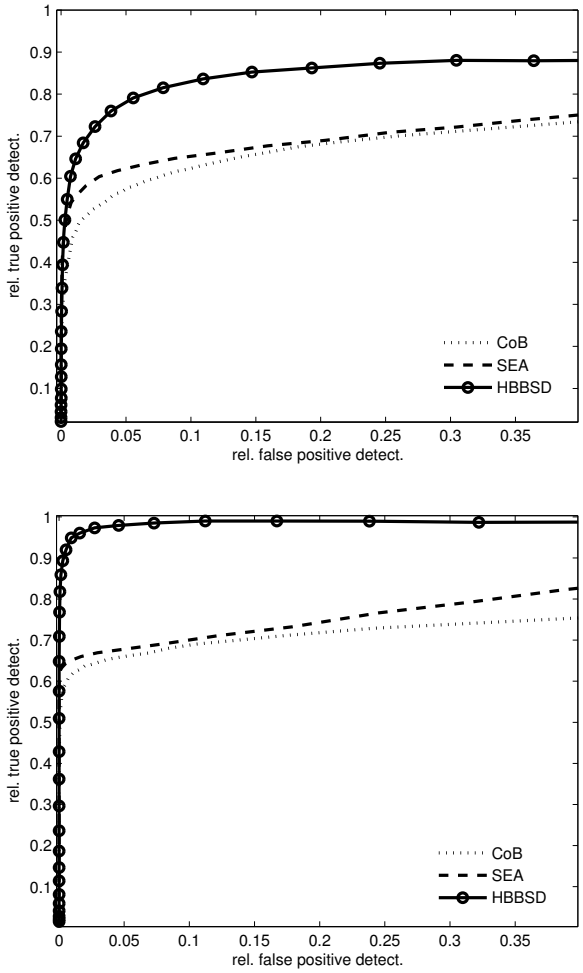

Fig. 5. Average ROC curves over 10 simulations for various spike detections methods. Top: Results for a dataset with SNR = 3.0. Bottom: Results for a dataset with SNR = 4.0.

E. Performance on data with two neurons

The second data set contained activity from two neurons, spiking at frequencies of 15Hz and 25Hz respectively, and the corresponding waveforms are shown in Fig. 2. The SNR was varied from 3.0 to 4.25 in steps of 0.25 (both spike trains always had equal SNR values), and again the ROC curves were computed for every method, see Fig. 5. To assess the overall performance for various SNR levels, the area under the ROC curves (AUC) was evaluated and is reported in Fig. 6. To summarize, the proposed method *HBBSD* always outperforms the other spike detection algorithms. In fact, *CoB* and *SEA* show poor performance, since these methods estimate only a single filter, which fails to detect both waveforms accurately, regardless of the SNR level. On the

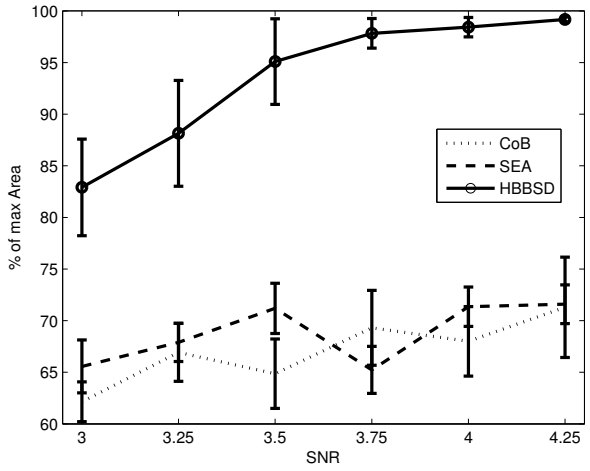

Fig. 6. Relative area under the ROC curves for various spike detection methods in dependence of the SNR level of the dataset. The bars indicate the standard deviations across the 10 simulations.

other hand, *HBBSD* estimates multiple filters, leading to an improved detection performance.

IV. Conclusion

To the knowledge of the authors, blind equalization algorithms relying on higher order statistics have rarely been applied to the task of neural spike detection. In this work, the super-exponential algorithm has been used for initial filter estimation. Furthermore, a mode detection and a sparse deflation procedure have been proposed in order to extract multiple spike waveforms, which have been then used for constructing MVDR filters.

To sum up, a novel method for unsupervised spike detection has been presented, which relies on the inherent characteristics of data from neural recordings, such as sparseness and finite alphabet. For instance, the sparseness of the neuronal signal was exploited for mode finding in the filter output and for proposing a sparse deflation procedure. On the other hand, the finite alphabet property allowed for an appropriate choice of the statistics for the SEA algorithm as well as for an easy estimation of the waveforms and construction of the MVDR filters. The main advantage of our method, namely that several data driven filters are calculated, resulted in a superior performance of *HBBSD* compared to wavelet methods or other existing blind filtering algorithms. This was demonstrated by a performance comparison on simulated datasets. Preliminary testing on real data seems to confirm these findings, and results will be presented in the near future. In order to allow for further evaluation, and application to real experiments, the proposed algorithm is being made available online to the community at the website http://user.cs.tu-berlin.de/∼natora/ .

Acknowledgment

This research was supported by the Federal Ministry of Education and Research (BMBF) with the grants 01GQ0743 and 01GQ0410. The authors would like to thank Prof. Aapo Hyvärinen for helpful discussion, Małgorzata M. Wójcik for help with proof-reading, and the anonymous reviewers for their suggestions.

References

[1] I. Obeid and P. D. Wolf, "Evaluation of spike-detection algorithms for a brain-machine interface application." *IEEE Trans Biomed Eng*, vol. 51, no. 6, pp. 905–911, Jun 2004.

[2] S. Mukhopadhyay and G. C. Ray, "A new interpretation of nonlinear energy operator and its efficacy in spike detection." *IEEE Trans Biomed Eng*, vol. 45, no. 2, pp. 180–187, Feb 1998.

[3] J. H. Choi, H. K. Jung, and T. Kim, "A new action potential detector using the mteo and its effects on spike sorting systems at low signal-to-noise ratios." *IEEE Trans Biomed Eng*, vol. 53, no. 4, pp. 738–746, Apr 2006.

[4] K. H. Kim and S. J. Kim, "A wavelet-based method for action potential detection from extracellular neural signal recording with low signal-to-noise ratio." *IEEE Trans Biomed Eng*, vol. 50, no. 8, pp. 999–1011, Aug 2003.

[5] Z. Nenadic and J. W. Burdick, "Spike detection using the continuous wavelet transform." *IEEE Trans Biomed Eng*, vol. 52, no. 1, pp. 74–87, Jan 2005.

[6] S. Dandapat and G. C. Ray, "Spike detection in biomedical signals using midprediction filter." *Med Biol Eng Comput*, vol. 35, no. 4, pp. 354–360, Jul 1997.

[7] S. Kim and J. McNames, "Automatic spike detection based on adaptive template matching for extracellular neural recordings." *J Neurosci Methods*, vol. 165, no. 2, pp. 165–174, Sep 2007.

[8] S. Shahid and L. S. Smith, "A novel technique for spike detection in extracellular neurophysiological recordings using cepstrum of bispectrum," *Proceedings of the 16th European Signal Processing Conference (EUSIPCO-08)*, 2008.

[9] A. Zviagintsev, Y. Perelman, and R. Ginosar, "Algorithms and architectures for low power spike detection and alignment," *Journal of Neural Engineering*, vol. 3, pp. 35–42, 2006.

[10] H. L. V. Trees, *Detection, Estimation, and Modulation Theory Part IV - Optimum Array Processing*. JOHN WILEY & SONS, 2002.

[11] C. M. Gray, P. E. Maldonado, M. Wilson, and B. McNaughton, "Tetrodes markedly improve the reliability and yield of multiple single-unit isolation from multi-unit recordings in cat striate cortex." *J Neurosci Methods*, vol. 63, no. 1-2, pp. 43–54, Dec 1995.

[12] M. Sahani, J. S. Pezaris, and R. A. Andersen, "On the separation of signals from neighboring cells in tetrode recordings," in *NIPS '97: Proceedings of the 1997 conference on Advances in neural information processing systems 10*. Cambridge, MA, USA: MIT Press, 1998, pp. 222–228.

[13] C. Pouzat, O. Mazor, and G. Laurent, "Using noise signature to optimize spike-sorting and to assess neuronal classification quality." *J Neurosci Methods*, vol. 122, no. 1, pp. 43–57, Dec 2002.

[14] C.-Y. Chi, C.-Y. Chen, C.-H. Chen, and C.-C. Feng, "Batch processing algorithms for blind equalization using higher-order statistics," *IEEE Signal Processing Magazine*, vol. 20, no. 1, pp. 25–49, Jan. 2003.

[15] O. Shalvi and E. Weinstein, "Super-exponential methods for blind deconvolution," *IEEE Transactions on Information Theory*, vol. 39, no. 2, pp. 504–519, March 1993.

[16] P. Paeaejaervi, "Adaptive blind deconvolution using third-order moments: Exploiting asymmetry," Ph.D. dissertation, Lulea University of Technology, Sweden, 2005, licentiate Thesis.

[17] M. A. Carreira-Perpinan, "Mode-finding for mixtures of gaussian distributions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1318–1323, Nov. 2000.

[18] M. C. Jones, J. S. Marron, and S. J. Sheather, "A brief survey of bandwidth selection for density estimation," *Journal of the American Statistical Association*, vol. 91, no. 433, pp. 401–407, 1996.

[19] Y. Inouye and K. Tanebe, "Super-exponential algorithms for multichannel blind deconvolution," *IEEE Transactions on Signal Processing*, vol. 48, no. 3, pp. 881–888, March 2000.

[20] M. Natora, F. Franke, M. Munk, and K. Obermayer, "Blind source separation of sparse overcomplete mixtures and application to neural recordings," *Lecture Notes in Computer Science - Independent Component Analysis and Signal Separation*, vol. 5441, pp. 459–466, 2009.