Published in final edited form as:

Med Biol Eng Comput. 2009 September ; 47(9): 955–966. doi:10.1007/s11517-009-0451-2.

Optimizing the Automatic Selection of Spike Detection Thresholds Using a Multiple of the Noise Level

Michael Rizk and Patrick D. Wolf

Department of Biomedical Engineering, Duke University, Durham, NC 27708, USA

Abstract

Thresholding is an often-used method of spike detection for implantable neural signal processors due to its computational simplicity. A means for automatically selecting the threshold is desirable, especially for high channel count data acquisition systems. Estimating the noise level and setting the threshold to a multiple of this level is a computationally simple means of automatically selecting a threshold. We present an analysis of this method as it is commonly applied to neural waveforms. Four different operators were used to estimate the noise level in neural waveforms and set thresholds for spike detection. An optimal multiplier was identified for each noise measure using a metric appropriate for a brain-machine interface application. The commonly used root-mean-square operator was found to be least advantageous for setting the threshold. Investigators using this form of automatic threshold selection or developing new unsupervised methods can benefit from the optimization framework presented here.

Keywords

Threshold; Spike detection; Noise estimate; Neural signal processor; Brain-machine interface

1 Introduction

High channel count fully implantable neural data acquisition systems must perform data reduction before transmitting recorded neural data over a wireless link [17,29,33,36]. However, these systems have significant size and power constraints and are limited in the amount and complexity of processing that they can perform [16]. In addition, timing constraints must be imposed on systems that are used in a real-time application such as a brain-machine interface (BMI). Consequently, the use of amplitude thresholding for spike detection is especially attractive for these systems due to its relative simplicity. Voltage thresholding has been the method of choice for spike detection in recently developed implantable neural signal processors [17,29,33,36].

We have developed a 96-channel neural signal processor as part of an implantable BMI system [33]. The processor automatically sets thresholds for each channel. The threshold for a given channel is set to a multiple of an estimate of the noise level on that channel. We wanted to know the optimal multiplier and the optimal noise measure to use in this system. Several groups have used a multiple of some measure of the noise level as a threshold; however, a few different noise measures have been used. The most commonly used measure of noise is the root-mean-square (rms) of the data over a given window, essentially the standard deviation of the data [9,11,14,19,29,31,35,41]. Other noise measures that have been used are the mean deviation

[33,37] and the value of the data located at a given percentile within a ranked set of data [15, 32,40]. It must be noted that all of the noise measures mentioned above do not measure only the noise. In reality, they are calculated from the entire neural signal. The samples that fall within a given window may simply be noise but they may also be spikes. So, the distribution of the samples is dependent not only on the distribution of neural noise but also on the size, shape, and number of spikes that fall within the window.

In order to effectively perform automatic threshold selection using a multiple of a noise measure, we would like to select a single value of the multiplier and trust that it will work reasonably well for all channels – even across channels that have different firing rates and SNRs. So, we would like to know whether or not there is a relatively consistent optimal multiplier for a given noise measure. A multiplier that is near optimal across waveforms with varying characteristics would be suitable for all channels in a high channel count system such as ours.

This paper investigates the problem of automatically selecting a threshold for spike detection. Thresholds computed as multiples of four different noise measures are compared as SNR, firing rate, and window size are varied. In addition, this study investigates the selection of an optimal multiplier for each of these noise measures in neural waveforms that contain spikes from either one or two neurons. The work presented here is motivated by the very real problem of selecting a noise measure and a multiplier for use in the 96-channel neural signal processor described in [33]. While the results presented in this paper are specific to our particular method of spike detection, the concepts and optimization framework used to perform the analysis are applicable more generally.

2 Methods

2.1 Simulated Waveforms

We compared the performance of thresholds computed as multiples of four different noise measures on simulated realistic neural signal waveforms. In order to simulate waveforms, we first constructed two libraries, one containing noise waveforms and the other containing spike templates. The noise library contained 17 5-second segments each from different channels of cortical microelectrode recordings from a rat. Each of these segments had no visible spikes and was thus classified as noise. The spike template library consisted of 10 spike templates from rat and monkey data. Each template was the average of similarly shaped spikes extracted from a single channel. The templates are shown in figure 1. All recordings were made using the neural data acquisition system described in [33]. Briefly, this system had a sampling rate of 31.25 kHz and a passband of 500 Hz to 6 kHz. Each data point was an 8-bit sample. For each channel, the system computed the mean value on that channel and subtracted the mean from incoming samples. This resulted in essentially zero-mean data.

For each simulated waveform, a 2-second span from one of the segments contained in the noise library was randomly selected. A template was also randomly selected to be used as the spike waveform to be inserted at locations within the 2-second noise waveform. This method of simulating neural signals has been used widely throughout the spike detection literature [2,4, 5,7,24].

Each simulated waveform was described by two parameters – SNR and firing rate. The SNR was used to scale the template. We define SNR as

$$\text{SNR} = \frac{\text{peak amplitude of scaled template}}{\text{standard deviation of 2-second noise waveform}}$$

(See section 4.7.1 for more on the definition of the SNR.) Spike times were selected to determine where in the noise waveform to place the spike waveforms. For a given firing rate, the appropriate number of spikes was placed into the waveform (for example, for a firing rate of 20 Hz there would be 40 spikes in a 2-second waveform). Spike times were randomly selected within the 2-second period; a refractory period of 1.6 ms was enforced. For each selected spike time, the spike waveform was added to the noise waveform (resulting in a noisy version of the “clean” spike) with the portion of the spike waveform with the highest amplitude placed at the selected spike time. Thus, the simulation resulted in realistic neural waveforms with known SNRs and spike times.

Rather than simulating an entirely new waveform (with a new noise waveform, a new spike template, and completely new spike times) for each combination of SNR and firing rate, sets of waveforms were simulated in a manner that limited all variation due to factors other than changing SNRs or firing rates. Each set of waveforms was generated in the following manner:

1. Randomly select a 2-second noise waveform.
2. Randomly select a spike template.
3. For the lowest firing rate of interest that has not yet been simulated, generate spike times. If spike times have already been generated for lower firing rates, start with those and generate more until the number of spikes corresponding to the current firing rate has been generated.
4. For each SNR of interest, use the noise waveform, spike template, and spike times to generate a full waveform.
5. If simulations have not yet been run for the highest firing rate of interest, go back to step 3.

The procedure described above generates neural waveforms that can be thought of as containing spikes from a single neuron. Commonly, signals from multiple neurons (or “units”) are observed on a single extracellularly recorded waveform. In order to simulate this circumstance, we generated two-unit waveforms. This was done by first simulating a single-unit waveform as described above. Then a second unit, with its own SNR and firing rate, was added to the waveform. The spike times for the second unit were selected independently of the spike times for the first unit. This created the potential for overlapping spikes from different units, a phenomenon often observed in real neural recordings. A template was scaled according to the SNR of the second unit and spikes were added into the waveform at the selected times regardless of if they overlapped with spikes from the first unit.

2.2 Noise Measures

For each simulated waveform, a sliding window was passed across the waveform. The sliding window was advanced one point at a time along the waveform. Each unique position of the sliding window will be referred to as an *epoch*. For each epoch, a noise measurement was made using four different operators – MD, RMS, P84, and PA68. These noise metrics are all based on measures that have been used previously in the neural spike detection literature for setting thresholds (see section 1). MD took the mean deviation of the points in the window. The simulated waveforms were essentially zero-mean due to the fact that the noise waveforms were nearly zero-mean (see section 2.1), the biphasic nature of the spike templates (allowing the spike templates to be close to zero-mean), and the fact that the spikes make up only a relatively small percentage of the overall waveform. Consequently, to compute MD we simply took the mean of the absolute value of the points in the window. RMS took the root-mean-square of all of the values in the window. P84 found the 84th percentile of the data in the window. This operator was based on a spike detection technique that has been implemented in a mixed-signal

neural signal processor [15, 40]. PA68 found the 68th percentile of the absolute value of the data in the window. This operator was chosen because it combines characteristics of the MD and P84 operators (taking the absolute value and finding a percentile, respectively). For zero-mean, normally distributed data, RMS, P84, and PA68 should approximate the standard

deviation of the data. MD should give $\sigma \sqrt{\frac{2}{\pi}}$, where σ is the standard deviation of the data [8]. The presence of high-amplitude spikes, however, causes neural data to be non-Gaussian [39]. Thus, the RMS, P84, and PA68 operators will not necessarily give identical results.

The MD and RMS operators were defined as follows:

$$\begin{aligned} \text{MD}(\mathbf{X}) &= \frac{1}{N} \sum_{i=1}^N |x_i| \\ \text{RMS}(\mathbf{X}) &= \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \end{aligned}$$

where N is the window size; the vector $\mathbf{X} = \{x_1, \dots, x_N\}$ contains the waveform samples that fall within the window; and $|\cdot|$ is the absolute value function. Percentiles for the P84 and PA68 operators were found using Matlab's *quantile* function.

2.3 Detection of Spikes

In order to perform spike detection, nonadaptive thresholds were applied to the absolute value of the data. This method of spike detection has been shown to be appropriate for computationally limited systems [28] and is used in the neural signal processor presented in [33]. Thresholds were computed as multiples of the noise measures. For each combination of noise measure and multiplier, a nonadaptive threshold was computed for each epoch and applied to the entire waveform. Thus, we applied tens of thousands of individual thresholds (one for each noise measure/multiplier pair for each epoch; each 2 s waveform has 62500 data points) to each simulated waveform. A refractory period of 1.28 ms (40 samples) was imposed so that a single spike would not cause multiple detections. Detections that occurred within 0.48 ms (15 samples) of the true spike time were considered to be accurate detections. Detections falling outside of this time window were considered to be false positives.

2.3.1 Probability of Detection and Rate of False Alarms—For each combination of noise measure and multiplier, we computed the thresholds, performed spike detection, and generated a pair of values to characterize the spike detection performance – the probability of detection (P_D) and the rate of false alarms (R_{FA}). The probability of detection is defined as

$$P_D = \frac{N_{\text{correct}}}{\text{actual number of spikes in the waveform}}$$

and the rate of false alarms is defined as

$$R_{FA} = \frac{\text{total number of detections} - N_{\text{correct}}}{\text{temporal duration of waveform}}$$

where N_{correct} is the number of correctly detected spikes. Note that while P_D is unitless, R_{FA} has units of false alarms per second. The values of P_D and R_{FA} are computed as averages across

the performance using thresholds computed for each epoch (see section 2.3). By using many different multipliers (and consequently many different thresholds), we were able to generate full receiver operating characteristic (ROC) curves. In this study, we used 31 multipliers ranging from 0 to 15 in steps of 0.5.

2.3.2 Optimality Index—In order to determine the optimal multiplier for each noise measure, we devised an optimality index (OI). The optimality index is a function of the probability of detection and the rate of false alarms, so each point on the ROC curve has a corresponding optimality index. As mentioned earlier, one application in which amplitude thresholding is a suitable means of spike detection is a fully implantable neural data acquisition system. One concern in such systems is the limited bandwidth available for transmitting data recorded inside the body to an external device. In these systems, false alarms can have a true cost by increasing latencies and preventing true spikes from being transmitted [6,33]. We formulated the optimality index using parameters suitable for this type of application.

Our formula for computing the optimality index is based on the cost function used in [28]. We define the optimality index as

$$OI = P_D - w \cdot \frac{(r \cdot P_D + R_{FA}) \cdot n \cdot b}{BW}$$

where P_D is the probability of detection, w is a unitless weight, r is the expected firing rate per channel, R_{FA} is the rate of false alarms per channel, n is the number of channels, b is the number of bytes that will be transmitted for each detected spike, and BW is the available bandwidth, or data rate, of the data acquisition system. The second term of the formula represents the fraction of the available bandwidth that is needed to transmit information about detections (whether true spikes or false alarms) multiplied by the cost, w , of using all of the available bandwidth. Throughout this study, we used 96 channels for n , 48 bytes per spike for b , and 96000 bytes per second for BW . These values were taken from a 96-channel system that sends a channel number, timestamp, and extracted spike waveform for each spike detected and has a limited wireless data rate [33]. We also used a value of 50 spikes per channel per second for r and a value of 0.1 for w , based on values used in [28]. The value used for r is a reasonable average for a channel with one or two neurons. The value of w can be adjusted based on how costly false detections are deemed to be. This is discussed in section 4.6.

In this study, we computed the optimality index separately for each SNR. This allowed us to find an optimal multiplier for each noise measure for each SNR.

2.3.3 Optimality Score—We used the optimality score to find the optimal multiplier when all SNRs were taken into consideration together. For a given multiplier, the optimality score is a weighted average of the optimality indices at that multiplier and each of the SNRs.

$$\text{optimality score} = \sum_{\text{SNR}} w_{\text{SNR}} \cdot OI_{\text{SNR}}$$

where w_{SNR} is the weight used for a given SNR and OI_{SNR} is the optimality index computed at that SNR. Table 1 presents the weights that we used. These weights correspond to the frequency with which waveforms of a given SNR were found in a sample real neural recording. The weights are based on the distribution of SNRs presented in figure 13 of [38]. That figure contains histograms that show the distribution of SNRs observed on electrodes implanted in the motor cortex for each of three monkeys.

It must be pointed out that SNR is defined differently in this paper and in [38]. In [38], SNR is defined as the peak-to-peak amplitude of the spikes divided by two times the standard deviation of the noise. The spike templates used in our study had, on average, peak-to-peak amplitudes that were approximately 1.5 times the size of their peak amplitudes. In order to

account for this, we scaled the SNR values presented in [38] by $\frac{4}{3}$ so that they would correspond to the SNR values used here. Once this was done, we used the frequency with which waveforms of a given SNR were found in the recordings from [38] to determine the weights used in the optimality score calculation.

3 Results

3.1 Single-Unit Waveforms

We simulated ten sets of waveforms. Each set had spikes from a single unit with SNRs ranging from 1 to 15 in steps of 1 and firing rates ranging from 5 Hz to 50 Hz in steps of 5 Hz. Each set of waveforms was generated as described in section 2.1. On each waveform, we computed all four noise measures using window sizes of 512, 1024, 2048, 4096, and 8192 samples and performed spike detection as described in section 2.3.

For the ten waveforms with a given SNR (from 1 to 15) and firing rate 20 Hz, the probability of detection and rate of false alarms values were used to construct optimality index curves for each of the four noise measures. These are plotted in figure 2 as a function of the multiplier value. The plots are based on spike detection performance using a window size of 4096 samples. The optimality indices were computed using overall probabilities of detection and rates of false alarms measured across ten waveforms with identical firing rates and SNRs. Curves are plotted for SNRs from 1 through 15. Figure 3(a) shows the maximum optimality index achieved for each noise measure at each SNR and figure 3(b) shows the optimality score for each multiplier for each noise measure. The multiplier that received the highest optimality score for each noise measure is listed in table 2. Optimality scores were also calculated for the other combinations of firing rate and window size. Table 3 shows the maximum optimality score obtained for each noise measure for selected parameter combinations. Overall, for a given set of parameters, comparable maximum optimality scores were achieved by all of the noise measures for the single-unit waveforms.

3.2 Two-Unit Waveforms

We simulated two-unit waveforms in which the first unit had an SNR of 8 and each unit had a firing rate of 20 Hz. For each set of waveforms, the SNR of the second unit was varied from 1 to 8 in steps of 1. Ten of these sets were simulated. Similar to section 3.1, spike detection was performed and optimality indices and scores were computed. These results are shown in figure 4. Since the SNR of the first unit was held constant at a value of 8, all variations in SNR referred to in the plots and calculations correspond to the SNR of the second unit.

Similarly, we also simulated two-unit waveforms in which the first unit had an SNR of 12. The results for this case are shown in figure 5.

3.3 Application of Thresholds to Real Neural Data

The key difference between applying thresholds to simulated data and to real neural data is that the true spike times are known for the simulated data while the true spike times are not known for the real data. As a result, analyses evaluating the performance of the threshold (or any other spike detection technique) cannot be performed on real data as they can be on simulated data. Nevertheless, it is useful to visualize the application of a given spike detection technique to real neural data. We took rat neural data recorded using the neural data acquisition

system mentioned in section 2.1 and computed thresholds in the manner used to compute thresholds for the simulated data. For each combination of noise measure and multiplier, an individual threshold was computed for each epoch. Thus, for each noise measure/multiplier pair, a distribution of thresholds was generated for each neural waveform. Figure 6 presents two examples of neural data along with the distributions of thresholds computed for selected combinations of noise measures and multipliers. To reduce clutter, the figure only shows negative thresholds. When applying a threshold to the absolute value of data, however, a positive threshold of equal magnitude would also be applied. The waveform shown in figure 6(a) appeared to have spikes from a single unit. The waveform shown in figure 6(b) appeared to have spikes from two distinct units.

4 Discussion

4.1 Modified Criteria for Evaluating Spike Detectors

Spike detectors are generally evaluated and compared based on performance either with a fixed set of parameters (for example, using a particular threshold) or across a range of parameters (resulting, for example, in an ROC curve). For certain types of systems, the performance of a detector (generally expressed in terms of metrics such as probability of detection and rate of false alarms) should not be looked at in isolation – other factors must be taken into account as well. Specifically, for high channel count systems with limited resources, such as implantable BMIs, it should be possible to set the parameters to satisfactory values relatively easily. These parameters ideally should be automatically determined. The usefulness of a detector that can perform very well when its parameters are set optimally but will perform poorly otherwise is limited by how reliably the correct parameters can be selected. Another factor that must be taken into account is the complexity of implementing the spike detector [28]. Developers of implantable neural signal processors have been willing to accept suboptimal spike detection performance compared to other types of spike detectors in return for the simplicity of implementing a voltage amplitude thresholding spike detection scheme. These systems already have high channel counts and the number of channels will only continue to increase in the future. As a result, the problem of automatically selecting thresholds is an important one. Spike detectors for these systems should be evaluated not only on their achievable levels of performance, but also on their complexity and ability to perform well using parameters that are selected without supervision.

4.2 Threshold Stability

Neural data is often thought of as consisting of large amplitude spikes superimposed on a noise cloud primarily representing background neural activity. Calculations that use a portion of the recorded data to estimate the noise level will be affected by spikes that happen to fall within the window over which the calculation is made. The level to which the calculation is affected depends on the firing rate and SNR of the spikes as well as the type of calculation.

A relatively straightforward way to automatically compute a threshold is to use a multiple of a noise measure as the threshold. In this case, having a relatively stable noise measure is desirable. Thresholds calculated based on the noise measure made during an initial monitoring period [33] ideally should not be significantly influenced by the location in time at which the monitoring period occurs (since that is essentially random).

4.3 Optimal Multipliers for Single-Unit Waveforms

Having to select a multiplier for each channel individually essentially defeats the purpose of trying to automatically set thresholds. If an individual is required to look at each channel in order to choose the multiplier, he might as well directly set the threshold for each channel. It is valuable to know the range over which multipliers are near optimal for waveforms with

different firing rates and SNRs. This way a multiplier that can perform well across many channels can be selected.

There can still be some design flexibility in picking the multiplier. A value toward the high end of the range would limit false detections and a value toward the low end would increase the probability of detection. At the same time, it is desirable to have a wide range of multipliers that are near optimal so that performance will not suffer greatly as long as a multiplier near the optimal multiplier is used. We used the optimality index described in section 2.3.2 and computed in section 3.1 to compare the optimal multipliers for the different noise measures.

From figure 2 it is clear that for a given noise measure, the optimality index can reach its peak at slightly different multiplier values for different SNRs. As a result, a single optimal multiplier cannot be identified. Nevertheless, across SNRs the optimal multiplier does generally fall within a small range. These ranges are different for the different noise measures (table 2). In order to pick a single multiplier to be used across channels with a variety of SNRs, an optimality score that is computed using weights corresponding to the frequency with which a given SNR occurs is appropriate.

While the optimality index curves shown in figure 2 look relatively similar for the different noise measures, one noticeable difference is the range over which the individual curves remain near their peaks. Curves remain near their peaks over the widest range for the MD operator. The range is slightly narrower for the P84 and PA68 operators. The RMS operator has the narrowest range. A review of the literature shows that many different multipliers have been used [9,11,15,19,29,31,32,33,35,40,41]. In some cases, these may not have been optimal.

4.4 Effect of Multiple Units

While the spike detection performance of all four operators was quite similar for waveforms containing spikes from only one unit, a couple of interesting observations can be made regarding the performance for waveforms containing spikes from two units. First, when large amplitude spikes from one unit are present along with lower amplitude spikes from a second unit, the maximum optimality indices achieved by the RMS operator at the SNRs that are most common (SNRs of 3 through 6 according to table 1) are lower than those achieved by the other operators. (Compare figure 4(b) and figure 5(b) with figure 3(a).) Second, the optimality score curve for the RMS operator peaks farther to the left (at a lower multiplier value) in the two-unit case than it does in the single-unit case. This can be seen by comparing figure 3(b) and figure 5(c). For the other operators, the multiplier for which the optimality score is highest remained unchanged across all of the different parameter combinations and circumstances investigated. This suggests that a single multiplier may prove to be optimal more generally for the MD, P84, and PA68 operators than for the RMS operator.

4.5 Comparison of Noise Measures

The complexity of neural noise should not be underestimated. It has been observed to be colored and non-stationary [12,24]. The complexity of the neural waveforms over which noise calculations are performed is further complicated by the presence of spikes in addition to the noise. These spikes distort the distribution of the samples in the waveform since they contain a disproportionate number of high amplitude samples [39]. As a result, the noise level estimates (and consequently the thresholds) computed by the different noise measures should not automatically be assumed to lead to equal performance as they might when applied to large samples of perfectly Gaussian data. Since the statistical nature of the overall neural waveform is not simple or obvious, an experimental approach (using real data as opposed to a statistical model) to the problem under consideration is warranted.

While the overall performance of all four measures investigated in this study was fairly similar, the root-mean-square operator did exhibit a few disadvantageous characteristics. The range of multipliers over which a given optimality index curve remained near its peak was narrowest for the RMS operator (section 4.3). In the two-unit case, the RMS operator had lower maximum optimality indices and a less consistent optimal multiplier than the other operators (section 4.4). Additionally, performing the squaring and square root operations used by the RMS operator can be problematic in certain types of systems. This suggests that using a percentile operator may be preferable for a system utilizing a mixed-signal processor [15, 40] and using the mean deviation operator may be preferable for a system utilizing a digital processor with limited memory resources [33].

4.6 Limitations and Extensions

It should be noted that the results presented in section 3.1 and section 3.2 and discussed in section 4.3 and section 4.4 use a specific value of the weight w . This weight is related to the cost incurred by false detections. The effect of false detections can range from simply being a nuisance to causing a severe drop in performance depending on the circumstances. For example, in a wired recording system in which spikes are detected and extracted with essentially no limit on the number of spikes and in which a reliable spike sorting process will be performed (meaning that most of the false extractions will later be categorized as noise rather than as true spikes), false detections are undesirable but should have only a minor impact on the performance of the system. In this case, the emphasis should be on picking a multiplier that results in a high probability of detection.

In the case of a fully implantable neural data acquisition system with a limited data rate, false detections will take up valuable bandwidth and could possibly prevent some true spikes from being transmitted [33]. Even if spike sorting is performed after the extracted spikes are received outside of the body (again categorizing false detections as noise), some of the true spiking information may be irretrievably lost due to the false detections. The cost of false detections might be relatively high in this case if the data rate necessary for true spikes is close to the available data rate.

Another case is a system that does not use spike sorting but instead simply processes spike detection information. This system would end up treating false detections exactly like true spikes. Thus false detections would never be removed and the performance of the system could be impacted negatively. In this case, multipliers that lead to many false detections would not be considered optimal.

From these examples it is clear that the multiplier that is considered to be optimal in one case may not be considered optimal in another. In this study, we looked at one possible case. From figure 6, we can see that the thresholds computed with the multipliers that were deemed to be optimal in this study fall very near the boundary of the noise cloud. It is likely that using these thresholds, some false detections will occur. Using larger multipliers would decrease the likelihood of false detections but may also result in a higher number of undetected true spikes.

This study, as all studies must, had a limited scope. Seemingly endless combinations of SNRs, firing rates, and window sizes could have been investigated, especially when allowing spikes from multiple neurons to be present in a given waveform. This study investigated only a limited subset of the possible parameter combinations. Nevertheless, many of the ideas and methods presented in this paper can be extended and applied to circumstances and spike detection techniques that do not exactly match those presented in this study. For example, we performed the same analyses described in this paper using adaptive thresholds rather than nonadaptive thresholds. The adaptive thresholds were constructed by setting the threshold at each point to a multiple of the noise measure computed over the epoch ending at that point. We found that

the results using adaptive thresholds were very similar to the average performance of nonadaptive thresholds presented here.

Also, this work is applicable beyond the realm of implantable neural data acquisition systems. Voltage thresholding is used widely to perform neural spike detection in many different contexts [9,11,14,15,17,18,19,23,26,29,30,31,32,33,35,36,37,40,41]. Other means of spike detection exist and have the potential to have improved spike detection performance [1,2,3,7,10,13,20,21,22,24,39]. Nevertheless, voltage amplitude thresholding, even with its less than ideal spike detection performance, is the predominant method of spike detection. This is in large part due to its simplicity and its ease of implementation. Automatic threshold selection is desirable in a wide variety of circumstances.

Finally, we suggest that the ability to easily and robustly select parameters that result in good performance across a wide range of realistic circumstances should be considered while developing and evaluating spike detection techniques. This is especially true for spike detection techniques intended for use in implantable systems. The optimization framework presented here may be useful in the development and evaluation of many of these techniques.

4.7 Final Notes

4.7.1 SNR—No consistent definition of SNR is found in the neural spike processing literature. Two commonly used definitions are the one used in this paper (see section 2.1) [7, 24, 41] and one in which the numerator of the ratio is the rms value of the template rather than the peak amplitude of the template [2, 9, 25, 34, 39]. Other definitions are used as well [5, 10, 21, 22, 38]. A definition of SNR that relies on the peak amplitude fits well with a detection scheme that uses an amplitude threshold since only the high amplitude portion of the spike is likely to cross the threshold and lead to a detection. Also, SNR computations that rely on the rms value of a spike are influenced by the assumed temporal duration of the spike. In general, exactly what portion of a full neural waveform should be considered to be part of a spike is not well-defined. For a useful note on SNR see [24].

4.7.2 Spike Detection Performance Parameters—In section 2.3.1, we characterize the spike detection performance by two parameters - P_D and R_{FA} . We defined R_{FA} as the number of false alarms divided by the temporal duration of the waveform. In the spike detection literature, it is more common to plot P_D versus the probability of false alarm (P_{FA}) or the “false alarm ratio” [5, 10, 18, 21, 24]. These are generally defined as the number of false alarms divided by the total number of detections. Since for realistic firing rates a single neuron’s spikes make up only a small percentage of the overall neural waveform, the number of false alarms is essentially independent of the firing rate. Rather, the number of false alarms is dependent on the noise characteristics and the spike detection parameters. The denominator of R_{FA} is also independent of the firing rate; thus, R_{FA} is essentially independent of the firing rate. On the other hand, the denominator of P_{FA} (namely, the sum of false alarms and true detections) is dependent on the firing rate. Consequently, P_{FA} can be very dependent on the firing rate. In a rough sense, for a given threshold, R_{FA} can be thought of as a measure of the noise level and P_D can be thought of as a measure of the signal level, both independent of firing rate. While plots of P_D versus R_{FA} do not exactly correspond to classical ROC curves, they do present the same general type of information. A few others have also plotted the probability of detection versus the rate of false alarms [22, 27].

5 Conclusion

We have investigated methods for automatically setting a spike detection threshold as a multiple of an estimate of a given channel’s noise level. The original motivation for this work was the problem of performing spike detection in the implantable neural signal processor

presented in [33]. The results of this study confirm that the mean deviation operator is an appropriate noise measure for use within our implantable neural signal processor. The results also provide guidance regarding the selection of appropriate multipliers for this means of spike detection.

Acknowledgements

We would like to thank Dr. Iyad Obeid for his helpful comments. This work was supported by Grant Number F31EB007897 from the National Institute of Biomedical Imaging and Bioengineering.

References

1. Aksenova TI, Chibirova OK, Dryga OA, Tetko IV, Benabid AL, Villa AEP. An unsupervised automatic method for sorting neuronal spike waveforms in awake and freely moving animals. *Methods* 2003;30(2):178–187. [PubMed: 12725785]
2. Bankman IN, Johnson KO, Schneider W. Optimal detection, classification, and superposition resolution in neural waveform recordings. *IEEE Trans Biomed Eng* 1993;40(8):836–841. [PubMed: 8258452]
3. Bar-Hillel A, Spiro A, Stark E. Spike sorting: bayesian clustering of non-stationary data. *J Neurosci Methods* 2006;157(2):303–316. [PubMed: 16828167]
4. Benitez R, Nenadic Z. Robust unsupervised detection of action potentials with probabilistic models. *IEEE Trans Biomed Eng* 2008;55(4):1344–1354. [PubMed: 18390325]
5. Borghi T, Gusmeroli R, Spinelli AS, Baranauskas G. A simple method for efficient spike detection in multiunit recordings. *J Neurosci Methods* 2007;163(1):176–180. [PubMed: 17391772]
6. Bossetti CA, Carmenta JM, Nicolelis MAL, Wolf PD. Transmission latencies in a telemetry-linked brain-machine interface. *IEEE Trans Biomed Eng* 2004;51(6):919–924. [PubMed: 15188859]
7. Brychta RJ, Tuntrakool S, Appalsamy M, Keller NR, Robertson D, Shiavi RG, Diedrich A. Wavelet methods for spike detection in mouse renal sympathetic nerve activity. *IEEE Trans Biomed Eng* 2007;54(1):82–93. [PubMed: 17260859]
8. Burrows BL, Talbot RF. The mean deviation. *Math Gazette* 1985;69(448):87–91.
9. Chandra R, Optican LM. Detection, classification, and superposition resolution of action potentials in multiunit single-channel recordings by an on-line real-time neural network. *IEEE Trans Biomed Eng* 1997;44(5):403–412. [PubMed: 9125825]
10. 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 Trans Biomed Eng* 2006;53(4):738–746. [PubMed: 16602581]
11. Farshchi, S.; Pesterev, A.; Guenterberg, E.; Mody, I.; Judy, JW. An embedded system architecture for wireless neural recording. 3rd Int IEEE/EMBS Conf on Neural Eng; 2007. p. 327-332.
12. Fee MS, Mitra PP, Kleinfeld D. Variability of extracellular spike waveforms of cortical neurons. *J Neurophysiol* 1996;76(6):3823–3833. [PubMed: 8985880]
13. Gozani SN, Miller JP. Optimal discrimination and classification of neuronal action potential waveforms from multiunit, multichannel recordings using software-based linear filters. *IEEE Trans Biomed Eng* 1994;41(4):358–372. [PubMed: 8063302]
14. Guillory KS, Normann RA. A 100-channel system for real time detection and storage of extracellular spike waveforms. *J Neurosci Methods* 1999;91(1–2):21–29. [PubMed: 10522821]
15. Harrison, RR. A low-power integrated circuit for adaptive detection of action potentials in noisy signals; Conf Proc IEEE EngMed Biol Soc; 2003. p. 3325-3328.
16. Harrison RR. The design of integrated circuits to observe brain activity. *Proc IEEE* 2008;96(7):1203–1216.
17. Harrison RR, Watkins PT, Kier RJ, Lovejoy RO, Black DJ, Greger B, Solzbacher F. A low-power integrated circuit for a wireless 100-electrode neural recording system. *IEEE J Solid-State Circuits* 2007;42(1):123–133.
18. Hatsopoulos NG, Xu Q, Amit Y. Encoding of movement fragments in the motor cortex. *J Neurosci* 2007;27(19):5105–5114. [PubMed: 17494696]

19. Jensen W, Rousche PJ. On variability and use of rat primary motor cortex responses in behavioral task discrimination. *J Neural Eng* 2006;3(1):7–13.
20. Kim KH, Kim SJ. Neural spike sorting under nearly 0-dB signal-to-noise ratio using nonlinear energy operator and artificial neural-network classifier. *IEEE Trans Biomed Eng* 2000;47(10):1406–1411. [PubMed: 11059176]
21. Kim KH, Kim SJ. A wavelet-based method for action potential detection from extracellular neural signal recording with low signal-to-noise ratio. *IEEE Trans Biomed Eng* 2003;50(8):999–1011. [PubMed: 12892327]
22. Kim S, McNamnes J. Automatic spike detection based on adaptive template matching for extracellular neural recordings. *J Neurosci Methods* 2007;165(2):165–174. [PubMed: 17669507]
23. Leiser SC, Moxon KA. Responses of trigeminal ganglion neurons during natural whisking behaviors in the awake rat. *Neuron* 2007;53(1):117–133. [PubMed: 17196535]
24. Nenadic Z, Burdick JW. Spike detection using the continuous wavelet transform. *IEEE Trans Biomed Eng* 2005;52(1):74–87. [PubMed: 15651566]
25. Nenadic Z, Burdick JW. A control algorithm for autonomous optimization of extracellular recordings. *IEEE Trans Biomed Eng* 2006;53(5):941–955. [PubMed: 16686417]
26. Nicolelis MA, Dimitrov D, Carmena JM, Crist R, Lehew G, Kralik JD, Wise SP. Chronic, multisite, multielectrode recordings in macaque monkeys. *Proc Natl Acad Sci USA* 2003;100(19):11,041–11,046.
27. Obeid, I. Comparison of spike detectors based on simultaneous intracellular and extracellular recordings. 3rd Int IEEE/EMBS Conf Neural Eng; 2007. p. 410–413.
28. Obeid I, Wolf PD. Evaluation of spike-detection algorithms for a brain-machine interface application. *IEEE Trans Biomed Eng* 2004;51(6):905–911. [PubMed: 15188857]
29. Olsson RH III, Wise KD. A three-dimensional neural recording microsystem with implantable data compression circuitry. *IEEE J Solid-State Circuits* 2005;40(12):2796–2804.
30. Paninski L, Fellows MR, Hatsopoulos NG, Donoghue JP. Spatiotemporal tuning of motor cortical neurons for hand position and velocity. *J Neurophysiol* 2004;91(1):515–532. [PubMed: 13679402]
31. Pouzat C, Mazor O, Laurent G. Using noise signature to optimize spike-sorting and to assess neuronal classification quality. *J Neurosci Methods* 2002;122(1):43–57. [PubMed: 12535763]
32. Quiroga RQ, Nadasdy Z, Ben-Shaul Y. Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. *Neural Comput* 2004;16(8):1661–1687. [PubMed: 15228749]
33. Rizk M, Obeid I, Callender SH, Wolf PD. A single-chip signal processing and telemetry engine for an implantable 96-channel neural data acquisition system. *J Neural Eng* 2007;4(3):309–321. [PubMed: 17873433]
34. Rutishauser U, Schuman EM, Mamelak AN. Online detection and sorting of extracellularly recorded action potentials in human medial temporal lobe recordings, in vivo. *J Neurosci Methods* 2006;154(1):204–224. [PubMed: 16488479]
35. Snider RK, Bonds AB. Classification of non-stationary neural signals. *J Neurosci Methods* 1998;84(1–2):155–166. [PubMed: 9821647]
36. Sodagar AM, Wise KD, Najafi K. A fully integrated mixed-signal neural processor for implantable multichannel cortical recording. *IEEE Trans Biomed Eng* 2007;54(6):1075–1088. [PubMed: 17554826]
37. Soto E, Manjarrez E, Vega R. A microcomputer program for automated neuronal spike detection and analysis. *Int J Med Informatics* 1997;44(3):203–212.
38. Suner S, Fellows MR, Vargas-Irwin C, Nakata GK, Donoghue JP. Reliability of signals from a chronically implanted, silicon-based electrode array in non-human primate primary motor cortex. *IEEE Trans Neural Syst Rehabil Eng* 2005;13(4):524–541. [PubMed: 16425835]
39. Thakur PH, Lu H, Hsiao SS, Johnson KO. Automated optimal detection and classification of neural action potentials in extra-cellular recordings. *J Neurosci Methods* 2007;162(1):364–376. [PubMed: 17353053]
40. Watkins, PT.; Santhanam, G.; Shenoy, KV.; Harrison, RR. Validation of adaptive threshold spike detector for neural recording; Conf Proc IEEE Eng Med Biol Soc; 2004. p. 4079–4082.

41. Zumsteg ZS, Kemere C, O'Driscoll S, Santhanam G, Ahmed RE, Shenoy KV, Meng TH. Power feasibility of implantable digital spike sorting circuits for neural prosthetic systems. *IEEE Trans Neural Syst Rehabil Eng* 2005;13(3):272–279. [PubMed: 16200751]

Fig. 1. Spike templates used in the simulated neural waveforms. No vertical scale bar is given because the template amplitudes were scaled based on the SNR of the simulated waveform. For each template, the dotted line shows the zero-point of the vertical axis.

Fig. 2.

Optimality index versus multiplier value for each of the noise measures. The firing rate was 20 Hz and the window size was 4096 samples. In each plot, each curve corresponds to a different SNR. The lower SNRs are on the bottom of the envelope; the higher SNRs are on the top of the envelope. The leftmost arrow in each plot points to the curve corresponding to an SNR of 1. The rightmost arrow in each plot points to the curve corresponding to an SNR of 15. The insets are zoomed-in views of areas of interest.

Fig. 3.

(a) The maximum optimality index value achieved at each SNR. A curve is plotted for each of the four different noise measures. All four curves lie almost directly on top of each other. (b) The optimality score as a function of multiplier value for each of the four noise measures. The optimality score is a weighted average of the optimality index across all SNRs. The RMS, P84, and PA68 curves all lie almost directly on top of each other. The plots in this figure are based on the data presented in figure 2.

Fig. 4.

Optimality index and optimality score for a two-unit waveform in which the first unit has an SNR of 8, each unit has a firing rate of 20 Hz, and the window size is 4096 samples. (a) Optimality index versus multiplier value for each of the noise measures. In each plot, each curve corresponds to a different SNR (1 through 8) for the second unit. The leftmost arrow in each plot points to the curve corresponding to an SNR of 1. The rightmost arrow in each plot points to the curve corresponding to an SNR of 8. (b) The maximum optimality index value achieved at each SNR of the second unit. (c) The optimality score as a function of multiplier value for each of the four noise measures. The SNR for the second unit was used when computing the optimality score.

Fig. 5.

Optimality index and optimality score for a two-unit waveform in which the first unit has an SNR of 12, each unit has a firing rate of 20 Hz, and the window size is 4096 samples. (a) Optimality index versus multiplier value for each of the noise measures. In each plot, each curve corresponds to a different SNR (1 through 12) for the second unit. The leftmost arrow in each plot points to the curve corresponding to an SNR of 1. The rightmost arrow in each plot points to the curve corresponding to an SNR of 12. (b) The maximum optimality index value achieved at each SNR of the second unit. (c) The optimality score as a function of multiplier value for each of the four noise measures. The SNR for the second unit was used when computing the optimality score.

(a)

(b)

Fig. 6.

Examples of real neural data along with the thresholds computed using various noise measures and multipliers. Histograms are presented showing the distribution of thresholds computed over all epochs using specific combinations of noise measures and multipliers. The multiplier is shown in parentheses next to the noise measure. The histograms show the locations of the computed negative thresholds. Zoomed-in views of the histograms are shown in the boxes below the histograms. (a) The waveform shows 2 s of neural data recorded from a rat. This waveform appears to have spikes from one discernible unit. (b) The waveform shows 2 s of neural data recorded from a rat. This waveform appears to have spikes from two discernible units of noticeably different amplitudes.

Table 1

Weights used for each SNR in computing the optimality score. SNRs that do not appear in this table were given a weight of zero.

SNR	Weight
2	0.05
3	0.15
4	0.31
5	0.22
6	0.13
7	0.07
8	0.04
9	0.02
10	0.01

Table 2

Comparison of optimal multipliers across the different noise measures. The second column gives the multiplier with the highest optimality score. The third column gives the range in which the highest optimality index is typically obtained when looking at different SNRs individually.

Noise Measure Operator	Overall Optimal Multiplier	Range of Typical Optimal Multipliers
MD	4.5	4.5...6.0
RMS	3.5	3.5...4.5
P84	3.5	3.5...5.0
PA68	3.5	3.5...5.0

Maximum optimality score obtained for each operator for several combinations of firing rate and window size. In each of these cases, the multiplier that achieved the maximum was 4.5 for the MD operator and was 3.5 for each of the other three operators.

Firing Rate (Hz)	Window Size (samples)	MD	RMS	P84	PA68
5	512	0.5309	0.5275	0.5135	0.5218
5	4096	0.5381	0.5399	0.5370	0.5368
5	8192	0.5380	0.5402	0.5392	0.5375
20	512	0.5424	0.5394	0.5245	0.5329
20	4096	0.5528	0.5574	0.5487	0.5486
20	8192	0.5534	0.5581	0.5508	0.5500
50	512	0.5455	0.5418	0.5365	0.5442
50	4096	0.5573	0.5611	0.5586	0.5585
50	8192	0.5576	0.5619	0.5594	0.5591