Evaluation of Spike-Detection Algorithms for a Brain-Machine Interface Application

Iyad Obeid* and Patrick D. Wolf, *Member, IEEE*

***Abstract*—Real time spike detection is an important requirement for developing brain machine interfaces (BMIs). We examined three classes of spike-detection algorithms to determine which is best suited for a wireless BMI with a limited transmission bandwidth and computational capabilities. The algorithms were analyzed by tabulating true and false detections when applied to a set of realistic artificial neural signals with known spike times and varying signal to noise ratios. A design-specific cost function was developed to score the relative merits of each detector; correct detections increased the score, while false detections and computational burden reduced it. Test signals both with and without overlapping action potentials were considered. We also investigated the utility of rejecting spikes that violate a minimum refractory period by occurring within a fixed time window after the preceding threshold crossing. Our results indicate that the cost-function scores for the absolute value operator were comparable to those for more elaborate nonlinear energy operator based detectors. The absolute value operator scores were enhanced when the refractory period check was used. Matched-filter-based detectors scored poorly due to their relatively large compuational requirements that would be difficult to implement in a real-time system.**

***Index Terms*—Brain-machine interface (BMI), neural, spike detection.**

I. Introduction

PROTOTYPE brain machine interfaces (BMIs) have successfully used spike timing information from extracellular neural signals to actuate prosthetic devices [1], [2]. The advent of wireless data acquisition hardware will physically untether the BMI and facilitate fully implantable systems that will obviate the need for transcutaneous wires. Incorporating spike detection will allow the BMI to transmit only the action potential (AP) waveforms and their respective arrival times instead of the sparse, raw signal in its entirety. This compression reduces the transmitted data rate per channel, thus increasing the number of channels that may be monitored simultaneously [3]. Spike detection can further reduce the data rate if spike counts are transmitted instead of spike waveforms. Spike detection will also be a necessary first step for any future hardware implementation of an autonomous spike sorter [4].

This study evaluates a number of known spike-detection algorithms and determines which are best suited for hardware implementation in an application-specific integrated circuit (ASIC) with limited computational resources. A low power implantable ASIC for detecting and transmitting neural spikes will be an important building block for BMIs [5]. A hardware realization of a spike detector in a wireless BMI must operate in real-time, be fully autonomous, and function at realistic signal-to-noise ratios (SNRs).

In a typical spike detector, the signal is preprocessed to accentuate spikes and attenuate noise, and then passed through a threshold detector to determine spike locations [6]. Several spike detection and spike sorting algorithms rely on a simple voltage threshold with little or no preprocessing. Static detectors use either a single threshold to detect one edge [7] or a pair of thresholds to detect both rising and falling spike edges [8]. Adaptive thresholds contend with the changing background noise levels common to the nonstationary extracellular neural signal [9]. Although simple thresholding is attractive for real-time implementations because of its computational simplicity, it is thought to be sensitive to noise and often requires user input to set effective threshold levels [10]. Overlapping spikes further reduce the efficacy of simple threshold detectors [10].

A second class of preprocessor algorithms uses template matching. This method is particularly effective when the spike waveform to be detected is known *a priori*. Since this is rarely the case, the user must manually select spikes from a set of test data, which are then averaged to form a template [11]. Both template matching and simple thresholding perform poorly when spike signal to noise ratios are low [12]. The neural signal may also be filtered with a family of wavelets to extract details about signal energy in particular time-frequency windows [13]–[15]. The feature extraction properties of this technique make it particularly useful when the goal is to sort spikes as well as to detect them [13], [14]. Conversely, if the goal is only spike detection, then the multiple convolutions required for wavelet decomposition may be computationally prohibitive for a real-time multichannel detector.

Energy-based spike detectors have also been used to detect neural spikes [12], [16], [17]. The nonlinear energy operator (NEO), also called the Teager energy operator, first characterized by Kaiser, estimates the square of the instantaneous product of amplitude and frequency of a sufficiently sampled signal [18]. In this regard, the NEO may be considered superior to other energy estimators that simply average the square of the signal and are independent of frequency. Variations of the NEO have also been proposed to improve detection. Filtering the NEO signal with a Bartlett window reduces the error in the estimate of the signal energy [16]. A generic NEO is thought to be more robust in the presence of noise [19]. A combination of wavelet decomposition and the NEO has also been described [20]. NEO-based

Manuscript received June 19, 2003; revised February 6, 2004. This work was supported by the Defense Advanced Research Projects Agency (DARPA) Defense Sciences Office through the Space and Naval Warfare Systems Center, San Diego, CA, under Contract N66001-02-C-8022 *Asterisk indicates corresponding author.*

*I. Obeid is with the Department of Biomedical Engineering, Duke University, Durham, NC 27707 USA (e-mail: io@duke.edu).

P. D. Wolf is with the Department of Biomedical Engineering, Duke University, Durham, NC 27707 USA.

Digital Object Identifier 10.1109/TBME.2004.826683

0018-9294/04$20.00 © 2004 IEEE

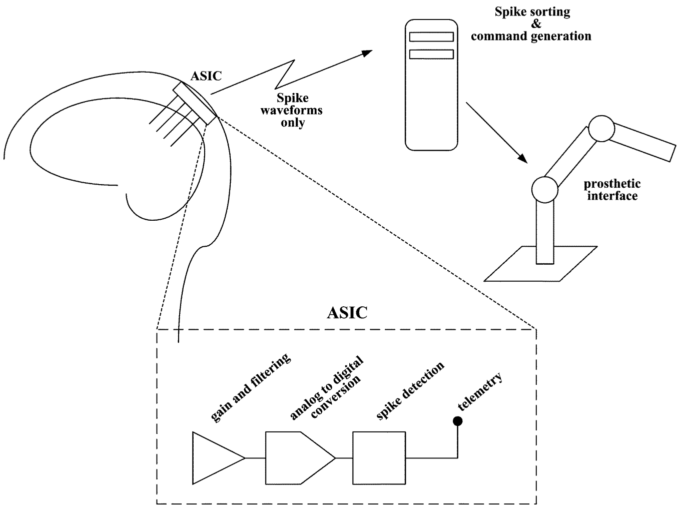
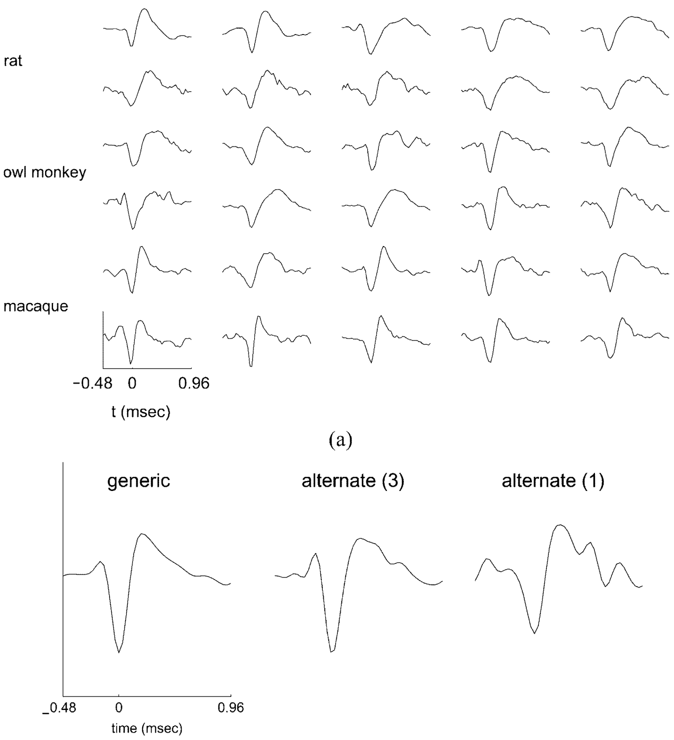


Fig. 1. Block diagram of the BMI under design. An implanted ASIC conditions signals from extracellular neural electrodes, digitizes them, and then detects AP spikes. The spike waveforms are transmitted across the skin to a BMI processor which sorts the spikes and then generates the command signals for the prosthesis.



(a)

(b)

Fig. 2. (a) The 30 AP waveforms used to generate the test data set. These were isolated from *in vivo* recordings of a rat, owl monkey, and macaque. (b) The three MF templates used to analyze the data. The *generic* template is averaged over all 30 base APs. The two *alternate* templates represent averages over three and one base AP, chosen at random from the original 30.

spike detection is attractive because of its ease of implementation and computational simplicity [16], [21].

II. Methods

We evaluated different preprocessors by applying them to a set of artificial test data and then sweeping threshold values. The numbers of correctly detected and false positive spikes were tabulated and entered into a cost function that scored the relative benefits of using each preprocessor in a generic BMI (Fig. 1). The BMI consists of an implantable ASIC that processes signals from n implanted neural electrodes, and then detects and transmits APs to a receiver that sorts the spikes, and uses them to control a prosthesis. Since false positive detections will be rejected by the sorter, they are detrimental only inasmuch as the telemetry bandwidth that they consume. The ideal spike detector must therefore maximize the number of correctly detected APs while limiting the number of false positive spikes that would otherwise reduce the effective system bandwidth.

A. Test Data Set

Two sets of 300 artificial test signals (duration 1 s, $F_S = 31.25$ kHz) were created using a bank of 30 AP waveforms isolated from *in vivo* motor cortex recordings from a rat, an owl monkey, and a macaque (ten APs per species) aligned at their minimum points [Fig. 2(a)]. Signals in the first test set consisted of one near-field "signal" neuron and 50 noise neurons. Each of the 30 APs was used as the signal neuron for a family of ten signals, with the signal neuron amplitudes being varied to achieve ten signal to noise ratios ranging linearly from 1.0 to 4.6. In every signal, the near-field neuron fired 50 times, once per 500 samples (16 ms). The firing rates of the 50 noise neurons were uniformly distributed between 50–90 Hz. The number of noise neurons is an approximation based on the assumptions that: 1) only neurons within 140 μm of the electrode are detectable [22]; 2) the density of motor cortex neurons in primates is 30 000/mm^3 [23]; 3) AP amplitudes are inversely proportional to their distance from the electrode; and 4) a portion of the detectable neurons may be silent during a particular task [22]. The combination of 50 noise neurons with their firing rate spread produces an average of $\sim$5 noise samples per time step, sufficient to create a realistic background signal. The calculated thermal noise contributions from a 1 MΩ recording electrode and an analog front end amplification circuit $(V_{\mathrm{th}} = 1\mu V_{\mathrm{rms}})$ were added to the signals. All signals were bandpass filtered with a linear phase FIR filter (450–6.5 kHz). A single family of signals from the first test data set are shown in Fig. 3.

The second test set consisted of signals with pairs of overlapping signal spikes and 50 noise neurons. Fifty pairs of spikes were located every 500 samples, with the interspike interval of each pair selected at random between 0 and 50 samples. The spike waveforms for both spikes in each pair were chosen at random from the bank of 30 *in vivo* waveforms. As before, 300 test signals were generated at ten SNRs varying linearly from 1.2 to 4.8.

B. Preprocessors

Three classes of preprocessors were considered.

1) Simple Threshold:

a) *Null*: No preprocessing—positive threshold is applied to raw data.

b) *Negation*: Signal is inverted before thresholding—equivalent to applying a negative threshold

c) *Absolute Value (Abs)*: Equivalent to applying both positive and negative thresholds simultaneously.

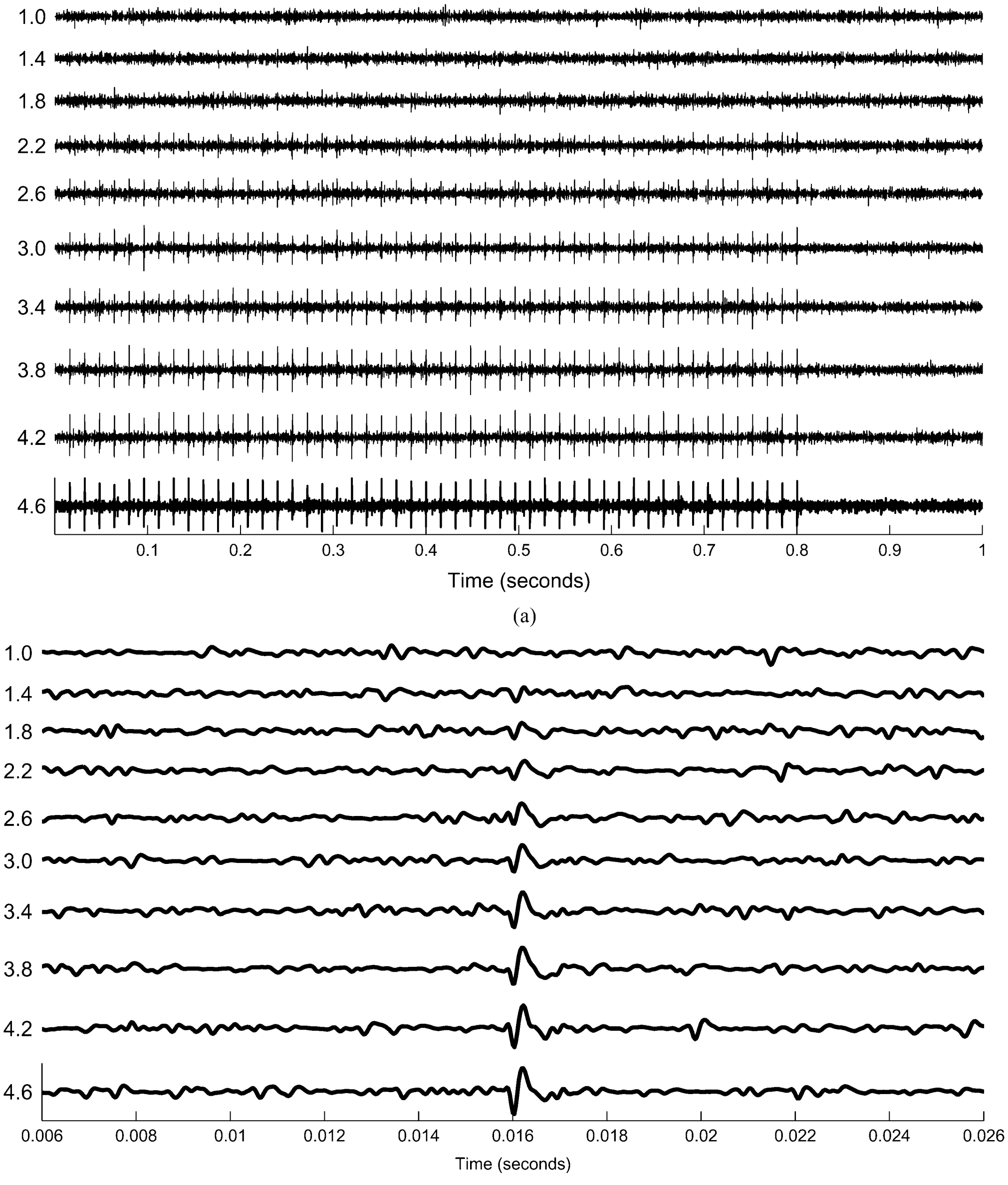


Fig. 3. A family of ten test signals generated using the same AP waveform as the sole "near-field" neuron. The amplitude of the near-field neuron is varied to create signals with ten evenly spaced SNRs. The SNR values for each signal are shown to the left. (a) The entire 1-s test signal is shown. (b) The signals are zoomed in, showing the first spike of each trace in detail.

2) *Energy Based:*

a) $\mathrm{NEO}(\delta)$: NEO, defined by $\Psi\{x[n]\} = x^2[n] - x[n+\delta]x[n-\delta]$, where $1 \leq \delta \leq 4$.

b) $\mathrm{SNEO}(\delta)$: Smoothed NEO, defined as the convolution of $\mathrm{NEO}(\delta)$ with a six point Bartlett window [16]; $1 \leq \delta \leq 4$.

3) *Matched-Filter (MF) Based:*

a) *MF*: The raw signal is convolved with a template that is the average of the 30 *in vivo* APs used to generate the test sets.

b) $\mathrm{NEO}(\delta) w/MF$: Convolution of the $\mathrm{NEO}(\delta)$ signal with a template that is the average of the AP waveforms after being processed by $\mathrm{NEO}(\delta)$.

c) *Absolute Value w/MF*: Convolution of *Abs* with a template that is the average of the AP waveforms after being processed by *Abs*.

There were a total of 17 preprocessors. The MF template was created by averaging together all 50 AP waveforms from each of the 30 largest signals in the first test set. Since the AP times were known precisely, there was no alignment error associated with averaging the APs from the artificial signals.

C. Spike Detection

The preprocessors were evaluated by applying them to all of the signals in the two test sets, and then sweeping a threshold across the resulting range of values, counting the positive threshold crossings as detections. Detections were tested both with and without a refractory period (RP) check that rejected positive threshold crossings occurring fewer than 20 samples apart ($t = 0.64$ ms). Each qualifying positive threshold crossing was classified as either a true or a false detection depending on whether it occurred within 15 samples ($t = 0.48$ ms) of an actual spike. The number of correct detections for each preprocessor was normalized by the actual number of APs (50 for nonoverlapping signals and 100 for overlapping ones) and averaged over the 30 test signals of equal SNR to compute the probability of detection, P_D. The number of false detections for each preprocessor was averaged over the 30 test signals of equal SNR to compute the number of false alarms, nFa.

D. Cost Function

A cost function was developed to evaluate the relative benefits of using each of the preprocessors in a generic BMI. The cost function is comprised of three terms, each normalized to one and weighted according to its perceived relevance

$$\begin{aligned} &CF(P_D, nFa, pp) \\ &= w_1 \cdot P_D - w_2 \frac{(r_{ms} \cdot m \cdot P_D + nFa) \cdot n \cdot b}{\mathrm{BW}} - w_3 \frac{C(pp) \cdot F_s \cdot n}{F_c} \end{aligned} \tag{1}$$

where $\boldsymbol{P_D}$ is the probability of correctly detecting a spike; $\boldsymbol{nFa}$ is the number of false alarm detections per second; $\boldsymbol{pp}$ is the preprocessor; $\boldsymbol{w_{1,2,3}}$ are the weights; $\mathbf{r_{ms}}$ is the maximum expected sustained firing rate per neuron; $\boldsymbol{m}$ is the maximum expected number of neurons per channel; $\boldsymbol{n}$ is the number of channels in the BMI; $\boldsymbol{b}$ is the number of bytes per spike transmitted by the BMI; $\boldsymbol{BW}$ is the available wireless data rate of the BMI; $\boldsymbol{C(pp)}$ is the number of clock cycles required by preprocessor pp to compute one value; $\boldsymbol{F_s}$ is the BMI sampling rate; and $\boldsymbol{F_c}$ is the system clock frequency. The values for $C(pp)$ are shown in Table I and are based on the assumption that each multiply and accumulate (MAC) operation in an ASIC requires ten clock cycles [24], while negations and squarings (which can be realized with look-up tables) require only one clock cycle each.

The three terms of (1) reflect the fact that an effective BMI must: 1) detect as many APs as possible; 2) not exceed its maximum wireless data rate; and 3) operate in real time. The second term reflects the bandwidth limit by forcing the maximum number of detected spikes per channel per second (both correct detections and false alarms) times the number of channels times the number of bytes per spike to be less than the available wireless data rate: $(r_{ms} \cdot m \cdot P_D + nFa) \cdot n \cdot b \leq \mathrm{BW}$, or equivalently, $((r_{ms} \cdot m \cdot P_D + nFa) \cdot n \cdot b)/(\mathrm{BW}) \leq 1$. The third term reflects the real-time implementation constraint by requiring that the total number of clock cycles per second $(C(pp) \cdot F_s \cdot n)$ be less than the clock frequency, or equivalently, $(C(pp) \cdot F_s \cdot n)/(F_c) \leq 1$. The second and third terms in (1) are subtracted so that the score is reduced as the system's available wireless bandwidth and clock cycles are consumed. The cost function was evaluated using constants that relate to the specific requirements of a BMI we are developing (Table II). The value for the bandwidth, BW, is the net data throughput measured in our laboratory across an IEEE 802.11-b wireless UDP network. The weights $w_{1,2,3}$ were selected to reflect our subjective interpretation of the relative importance of detecting as many APs as possible since even small numbers of missed spikes can greatly deteriorate the information content of a spike train [25].

TABLE I
ESTIMATED NUMBER OF CLOCK CYCLES REQUIRED PER SAMPLE TO PERFORM SPIKE DETECTION USING EACH OF THE PREPROCESSORS

Preprocessor	Req'd Computation	Req'd Clock Cycles
Null	*N/A*	0
Negation	1 neg	1
NEO(δ)	1 sq. & 1 MAC	11
SNEO(δ)	NEO(δ) & 5 MACs	61
Abs	1 neg.	1
Matched Filter	51 MACs	510
NEO(δ) w/ MF	NEO(δ) & 51 MACs	521
Abs w/ MF	1 neg. & 51 MACs	511

TABLE II
CONSTANT TERMS USED TO EVALUATE THE COST FUNCTION (1)

Constant	Value	Units
n	96	channels
b	70	bytes/spike
r_{ms}	50	spikes/sec
m	3	neurons/channel
F_s	40k	samples/sec
F_c	96×10^6	clock cycles/sec
BW	360k	bytes/sec
w_1	10	
w_2	1	
w_3	1	

III. RESULTS

Families of curves for four representative preprocessors are shown in Fig. 4; input signals are swept from minimum to maximum SNR. The x axis for each preprocessor has been normalized to the maximum range of the preprocessed signal. For any given preprocessor, signals with low SNRs score relatively poorly and have a limited range of threshold values that produce the best score. Signals with bigger SNRs result in greater scores over a larger range of threshold values. The maximum cost-function score is realized when all of the true APs and no false alarms are detected. When the threshold is at 100% of the input range, no spikes (correct or false) are detected, and the cost-function score reduces to just the third term of (1), which is a constant for each preprocessor. *Abs* produces the greatest cost-function scores; scores for the Negation and NEO operators are slightly lower than those for *Abs* and are maximized over a smaller range of threshold values. The MF works over the widest range of threshold values but produces a substantially lower cost-function score due to its computationally intensive nature.

In Table III, the maximum scores of the SNR curves are averaged for each preprocessor and ranked. For brevity, only the top five scores in each category (overlapping vs. nonoverlapping spikes; detection with RP versus without) are shown. For both overlapping and nonoverlapping data, the *Abs* preprocessor used with an RP is the best overall detector. The necessity of pairing *Abs* with an RP is as expected; without an RP, both phases of a biphasic AP would produce a valid positive threshold crossing, incurring one false alarm for every true detection. Excluding the RP improved scores slightly for

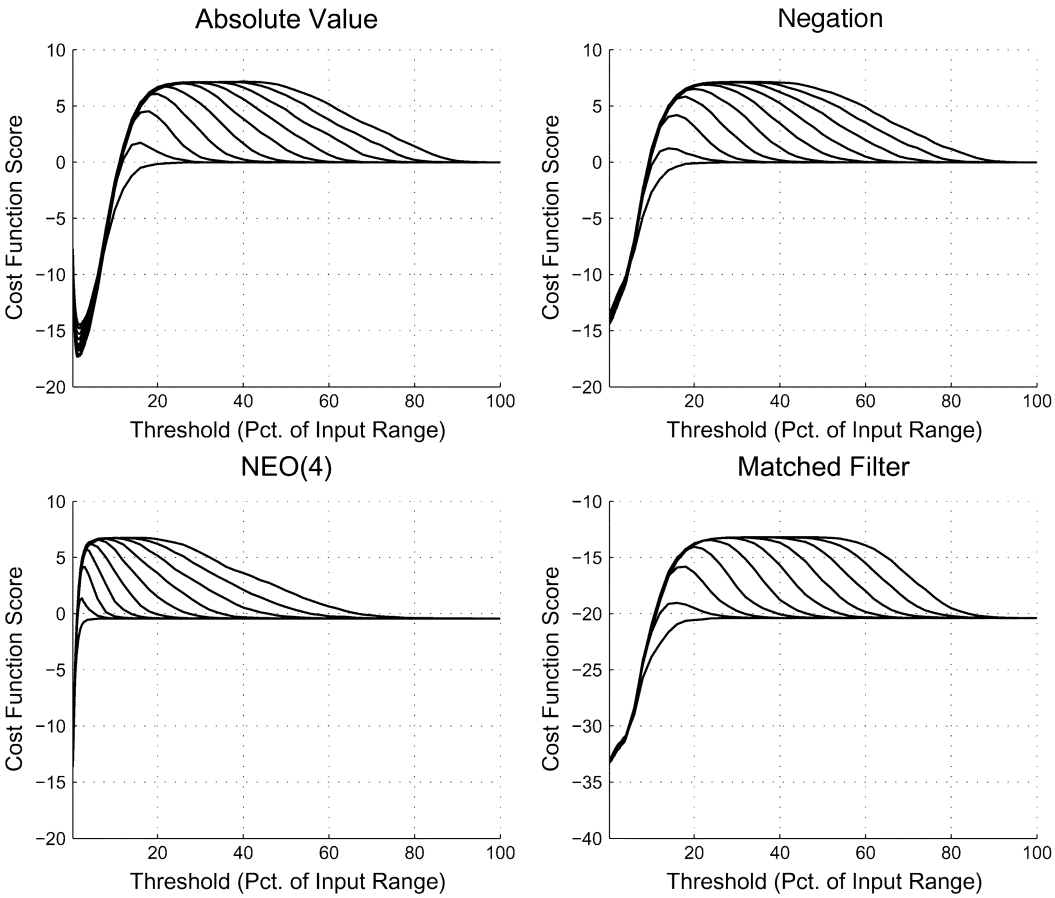


Fig. 4. Families of cost-function score curves for four preprocessors. Each of the ten curves within a family denotes a different input signal SNR. All curves were taken from data generated with nonoverlapping spikes and using a 20 sample refractory period to qualify positive threshold crossings as valid spikes. *Abs*, *Negation*, and NEO(4) are the best three preprocessors in this category (see Table III). The MF curves are included for comparison. Note that the y-axis for the MF plot does not match those of the other three. Cost-function scores for the MF are as low as they are because of the large number of MAC required for computation.

TABLE III
THE FIVE OVERALL HIGHEST SCORING PREPROCESSORS IN EACH OF FOUR CATEGORIES. COST FUNCTION SCORES ARE COMPUTED USING (1) AND THE CONSTANTS IN TABLE II

	No Overlap		Overlap	
	Preprocessor	Score	Preprocessor	Score
with Refr. Per.	Abs	5.46	Abs	2.80
	Negation	5.31	Negation	2.33
	NEO(4)	5.06	NEO(4)	2.29
	NEO(3)	5.01	NEO(2)	2.20
	Null	4.98	NEO(1)	2.19
w/o Refr. Per.	Negation	5.32	Abs	2.59
	NEO(3)	4.92	Negation	2.56
	Abs	4.92	Null	2.40
	NEO(2)	4.89	NEO(3)	2.31
	NEO(1)	4.83	NEO(2)	2.30

TABLE IV
THE FIVE HIGHEST SCORING PREPROCESSORS WHEN COMPUTATIONAL COMPLEXITY IS DISREGARDED IN COMPUTING THE COST FUNCTION SCORE [I.E., $w_3 = 0$ IN (1)]

	No Overlap		Overlap	
	Preprocessor	Score	Preprocessor	Score
with Refr. Per.	MF	5.52	Abs	2.84
	Abs	5.50	NEO(4)	2.73
	NEO(4)	5.50	NEO(4) w/ MF	2.68
	SNEO(4)	5.47	NEO(3) w/ MF	2.66
	NEO(3)	5.45	Abs w/ MF	2.66
w/o Refr. Per.	MF	5.53	SNEO(4)	2.76
	NEO(4) w/ MF	5.44	NEO(3)	2.75
	NEO(3) w/ MF	5.43	NEO(2)	2.74
	SNEO(3)	5.42	NEO(1)	2.73
	SNEO(4)	5.41	SNEO(3)	2.72

overlapping spikes. Table III also shows that the performances of the simple threshold and energy-based preprocessors are almost identical.

To assess preprocessors in systems that are not computationally limited, the data were reanalyzed by setting $w_3 = 0$ in (1) (Table IV). For nonoverlapping signals, MF-based operators perform comparably with energy-based operators, despite the nonspecific nature of the generic MF template. The MF did not perform as effectively for signals with overlapping spikes ($\text{score} = 2.39$ with RP; 2.68 without RP).

We compared the average SNR gain of the preprocessors to their relative cost-function scores and found no correlation (Table V). The simple threshold preprocessors (Null, Negation, Absolute Value) provide no SNR gain, yet score comparably to energy-based operators that have SNR gains of up to 5.2. The MF showed slightly negative gains as a result of the nonspecific nature of the generic template.

To quantify whether different preprocessors perform better at different SNRs, the averages of the best cost-function scores

TABLE V
SIGNAL TO NOISE RATIO GAIN (IN dB). THE GAINS FOR $\mathrm{NEO}(\delta)$, $\mathrm{SNEO}(\delta)$, AND $\mathrm{NEO}(\delta)$ w/MF WERE VERY SIMILAR, AND SO THEY WERE AVERAGED FOR BREVITY

Preprocessor	**Raw Signal SNR**				
	1.4	**2.2**	**3.0**	**3.8**	**4.6**
Simple Threshold	0.0	0.0	0.0	0.0	0.0
NEO	2.3	6.4	10.1	12.5	14.3
SNEO	2.3	6.8	10.1	12.5	14.3
MF	0.0	-0.9	-0.9	-0.9	-1.9
NEO w/ MF	2.3	6.0	9.2	10.9	11.8
Abs w/ MF	1.6	5.1	7.6	8.9	9.2

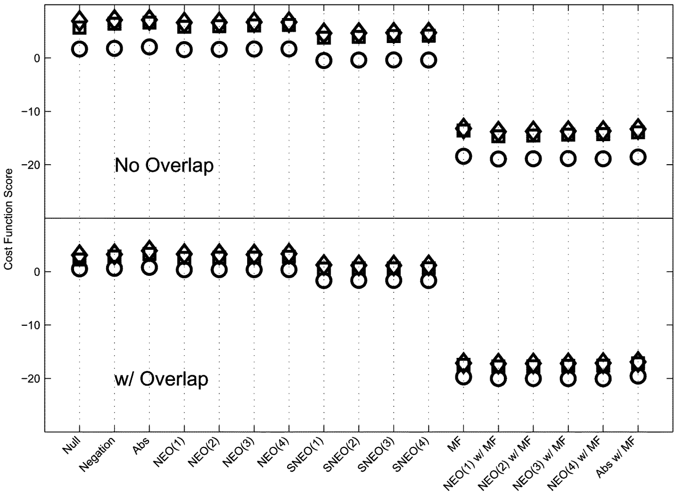


Fig. 5. Cost-function scores are averaged over three different ranges of input SNRs. ○ denotes $\mathrm{SNR} < 1.8$; □ denotes $2.6 \leq \mathrm{SNR} \leq 3.0$; ◇ denotes $\mathrm{SNR} \geq 3.4$. Scores are shown for data both with and without overlapping spikes. All detections were computed using the 20-point refractory period.

for small $(\mathrm{SNR} \leq 1.8)$, medium $(2.6 \leq \mathrm{SNR} \leq 3.0)$, and large $(\mathrm{SNR} \geq 3.4)$ signals were computed (Fig. 5). For signals both with and without overlapping spikes, *Abs* scored the best over all three ranges, although the NEO, Null, and Negation operators performed comparably.

IV. DISCUSSION

Although spike detection has been studied extensively, its implementation in a BMI presents new problems. BMI spike detectors must work in real-time using limited computational resources that are divided among dozens of channels. BMIs must also be fully autonomous; as the number of channels in a system scales, it will become unreasonable for a user to manually determine threshold levels or create spike templates. This study investigates how best to emphasize neural spikes against background noise, and does not address the issue of how to determine the optimal threshold value for a particular combination of preprocessor and input signal SNR. Our results indicate that the absolute value is as effective a preprocessor as any energy or MF-based operator. The data suggest that the best way of improving spike detection is not to employ an elaborate preprocessor, but rather to maximize SNR. This finding underscores the importance of minimizing circuit noise in the analog electronics that condition the neural signals before they are detected. One ramification of this work is that, since *Abs* is relatively simple to implement, an efficient spike detector can be realized in both the analog and digital domains, provided that there exists a realistic method for setting the threshold voltage autonomously. Our findings also demonstrate that SNR gain is not necessarily a good predictor of how well a preprocessor will emphasize spikes in a neural signal.

We compared the generic MF template with two others: one that was derived from only three of the 30 APs, and another derived from only one [Fig. 2(b)]. These were applied to the signals from the data sets that were based on the same AP waveforms as the ones included in the respective template. Only negligible differences in performance were found between the alternate templates and the generic one. Since performance did not depend on the precise shape of the template, it may be possible to design a MF spike detector that does not require a user to generate spike templates for each channel and have it perform as well as one that does.

A well-known energy-based preprocessor that has not been discussed in this study is the instantaneous square of the signal, $\psi\{x[n]\} = x^2[n]$. Since there is a one-to-one mapping between the absolute value and signal squared operators, there is theoretically no difference between the two for enhancing spike detectability. This was verified by simulation, and so these redundant results were omitted.

The cost function used in this study is a robust metric for evaluating spike detection in a BMI with limited computational resources. The constant terms in (1) may be modified to reflect the design constraints of different BMI architectures. For example, if a BMI under design will not include spike sorting in the receiver, then false positive detections will directly contribute to the prosthesis control, negatively affecting performance. In such a case, the weights w_1 and w_2 may be made equal to increase the importance of minimizing false positives relative to that of maximizing correctly detected spikes.

V. CONCLUSION

A technique has been described for comparing spike-detection algorithms for wireless BMIs. For systems with limited computational resources, taking the absolute value of the neural signal before applying a threshold (in combination with a refractory period) is just as effective for detecting spikes as applying more elaborate energy-based detectors. A novel cost function for comparing spike-detector performance has also been presented.

REFERENCES

[1] J. M. Carmena, M. A. Lebedev, R. E. Crist, J. E. O'Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. Henriquez, and M. A. L. Nicolelis, "Learning to control a brain-machine interface for reaching and grasping by primates," *PLoS Biol.*, vol. 1, no. 2, 2003.

[2] D. M. Taylor, S. I. Helms Tillery, and A. B. Schwartz, "Direct coritcal control of 3D neuroprosthetic devices," *Science*, vol. 296, pp. 1829–1832, 2002.

[3] K. A. Moxon, J. C. Morizio, J. K. Chapin, M. A. Nicolelis, and P. Wolf, "Designing a brain-machine interface for neuroprosthetic control," in *Neural Prostheses for Restoration of Sensory and Motor Function*, J. K. Chapin and K. A. Moxon, Eds. Boca Raton, FL: CRC Press, 2001.

[4] B. C. Wheeler, "Automatic discrimination of single units," in *Methods for Neural Ensemble Recordings*, M. A. Nicolelis, Ed. Boca Raton, FL: CRC Press LLC, 1999.

[5] M. A. L. Nicolelis, "Actions from thoughts," *Nature*, vol. 409, pp. 403–407, 2001.

[6] R. N. McDonough and A.D. Whalen, *Detection of Signals in Noise*, 2nd ed. San Diego, CA: Academic , 1995.

[7] S. N. Gozani and J. P. Miller, "Optimal discrimination and classification of neuronal action potential waveforms from multiunit, multichannel recordings using software-based linear filters," *IEEE Trans. Biomed. Eng.*, vol. 41, pp. 358–372, Apr. 1994.

[8] K. S. Guillory and R. A. Normann, "A 100-channel system for real time detection and storage of extracellular spike waveforms," *J. Neurosci. Meth.*, vol. 91, pp. 21–29, 1999.

[9] R. Chandra and L. M. Optican, "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.*, vol. 44, pp. 403–412, May 1997.

[10] M. S. Lewicki, "A review of methods for spike sorting; the detection and classification of neural action potentials," *Network*, vol. 9, no. 4, pp. R53–R78, 1998.

[11] I. N. Bankman, K. O. Johnson, and W. Schneider, "Optimal detection, classification, and superposition resolution in neural waveform recordings," *IEEE Trans. Biomed. Eng.*, vol. 40, pp. 836–841, Aug. 1993.

[12] K. Kim and S. Kim, "Neural spike sorting under nearly 0-dB signal-to-noise ratio using nonlinear energy operator and artificial neural-network classifier," *IEEE Trans. Biomed. Eng.*, vol. 47, pp. 1406–1411, Oct. 2000.

[13] X. Yang and S. Shamma, "A totally automated system for the detection and classification of neural spikes," *IEEE Trans. Biomed. Eng.*, vol. 35, pp. 806–816, Oct. 1988.

[14] E. Hulata, R. Segev, and E. Ben-Jacob, "A method for spike sorting and detection based on wavelet packets and shannon's mutual information," *J. Neurosci. Meth.*, vol. 117, pp. 1–12, 2002.

[15] K. Kim and S. Kim, "Wavelet-based action potential detector for the extracellular neural signal with low signal-to-noise ratio," presented at the *2nd Joint EMBS/BMES Conf.*, Houston, TX, 2002.

[16] 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, pp. 180–187, Feb. 1998.

[17] H. Park, D. Jeong, and K. Park, "Automated detection and elimination of periodic ECG artifacts in EEG using the energy interval histogram method," *IEEE Trans. Biomed. Eng.*, vol. 49, Dec. 2002.

[18] J. F. Kaiser, "On a simple algorithm to calculate the energy of a signal," presented at the *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Albuquerque, NM, 1990.

[19] R. Agarwal, J. Gotman, D. Flanagan, and B. Rosenblatt, "Automatic EEG analysis during long-term monitoring in the ICU," *Electroencephalogr Clin. Neurophysiol. Suppl.*, vol. 107, 1998.

[20] G. Calvagno, M. Ermani, R. Rinaldo, and F. Sartoretto, "A multiresolution approach to spike detection in EEG," presented at the *IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Istanbul, Turkey, 2000.

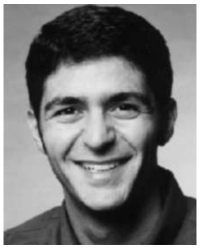
[21] M. Atit, J. Hagan, S. Bansal, R. Ichord, R. Geocadin, C. Hansen, D. Sherman, and N. Thakor, "EEG burst detection: Performance evaluation," presented at the *1st Joint BMES/EMBS Conf.*, Atlanta, GA, 1999.

[22] D. A. Henze, S. Borhegyi, J. Csicsvari, A. Mamiya, K. D. Harris, and G. Buzsaki, "Intracellular features predicted by extracellular recordings in the hippocampus *in vivo*," *J. Neurophysiol.*, vol. 84, pp. 390–400, 2000.

[23] M. Abeles, *Corticonics*. Cambridge, MA: Cambridge Univ. Press, 1991.

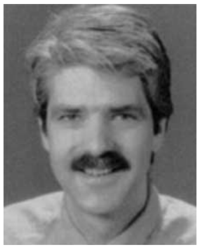
[24] S. Sunder, F. El-Guibaly, and A. Antoniou, "Two's-complement fast serial-parallel multiplier," in *Proc. Inst. Elect. Eng.—Circuits Devices Syst.*, vol. 142, 1995.

[25] D. S. Won, D. Y. Chong, and P. D. Wolf, "Effects of spike sorting error on information content in multi-neuron recordings," presented at the *1st Int. IEEE EMBS Conf. Neural Engineering*, Capri Island, Italy, 2003.



Iyad Obeid was born in Sheffield, U.K., in 1975. He received the B.S. and M.Eng. degrees from the Massachusetts Institute of Technology, Cambridge, in 1997 and 1998, respectively. He is currently working toward the doctorate degree in biomedical engineering at Duke University, Durham, NC, where he is investigating data acquisition issues for brain-machine interfaces.

Mr. Obeid is a 1999 recipient of the National Defense Science and Engineering Graduate Fellowships.



Patrick D. Wolf (M'98) was born in Altoona, PA, in 1956. He received the B.S. degree in electrical engineering and the M.S. degree in bioengineering, both from the Pennsylvania State University, University Park, and the Ph.D. degree from Duke University, Durham, NC, in 1992.

He then joined the faculty in Biomedical Engineering at Duke University, where he is pursuing his research interests in instrumentation, cardiac arrhythmias, and the brain-machine interface.