A Hybrid NEO-based Spike Detection Algorithm for Implantable Brain-IC Interface Applications

Anh Tuan Do and Kiat S. Yeo.
VIRTUS, School of EEE, Nanyang Technological University, Singapore 639798

***Abstract*—Real time spike detection is the first critical step to develop spike-sorting for brain-IC interface applications. For implantable VLSI implementation, spike detection hardware must consume low power and at the same time ensure high true positive probability while having as few false alarms as possible. Currently, Nonlinear Energy Operator (NEO) and absolute thresholding are the two most widely used spike detection algorithms where NEO has a slightly better performance measured by areas under the receiver operating characteristic (ROC) curves. This paper revisits these two algorithms and quantitatively points out that NEO is in fact much better than absolute thresholding. We also propose a hybrid algorithm that offers similar accuracy as NEO but only requires 11% of power consumption.**

I. INTRODUCTION

Multi-electrode intracranial recording technology offers exceptionally high spatial and temporal signal resolutions needed for neural prosthetic development and neuroscience research [1, 2]. Conventionally, analog signals from different neurons surrounding electrodes are digitized transmitted to a nearby computer for subsequent software processing such as spike sorting and neural encoding. However this approach shows limitation as power consumption and bandwidth of the transmitter becomes too large when number of electrodes approaches a few hundreds. For example, data rate for 100 channels, 25kS/s recording system is 20 Mb/s, assuming 8 bit ADC. Furthermore, neural prosthetic application dictates that software-based processing must be done online so that brain commands can be performed with negligible time delay. Engineers and scientists therefore have turned to hardware implementation to reduce the data rate before transmitting it to the subsequent software encoding state [1, 3]. This process is called spike sorting, where data is transmitted only when a spike is detected from the recorded signal. Fig. 1 illustrates a typical recording system where a multichannel analog front end receives brain signals directly from a 2D electrode array. After digitizing, brain signals are sent to on-chip, off-chip or software-based spike sorting processor. Spike sorting consists of several stages such as spike detection, alignment, feature extraction and clustering [4], also shown in Fig. 1.

This work focuses on spike detection algorithm. For implantable VLSI spike sorting applications, spike detection block must consume as small power as possible. We review the two most popular spike detection methods: Absolute Value and NEO. A new performance indicator (P-curve) is constructed to compare these two algorithms. Finally, we propose a hybrid method to deliver high detection accuracy while consuming very low power consumption for multichannel neural recording implementations.

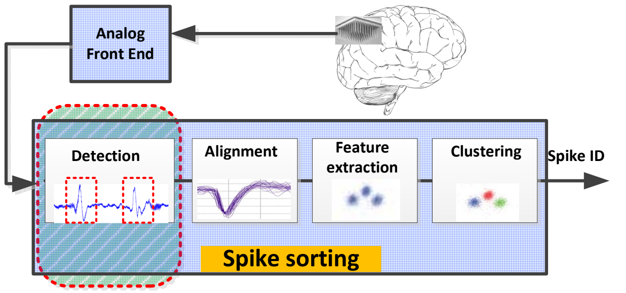

Figure 1. A typica brain signal recording system consists of a analog front-end and spike sorting back end.

II. SPIKE DETECTION ALGORITHMS

Spike detection consists of two stages namely pre-emphasizing and thresholding. Pre-emphasizing can come in the form of calculating the absolute, first derivative, second derivative or digital filter of the original neural recording signal. A predefined threshold will then be applied to the new data to determine the presence of spikes. There are certain trade-off between hardware complexity, power consumption and accuracy performance between these algorithms [5]. Until now, *Absolute Value* and *NEO* are the two most popular algorithms because of their simple hardware and high level of accuracy. We investigate these two algorithms as they are the best candidates for low-power VLSI implementation.

A. Absolute value

This is the most simple and commonly used detection method. A predefined threshold (*Thres_data*) is applied to the raw (or absolute of) recording data and a spike is identified if data point is higher than the threshold value. [6] confirmed that taking the absolute of raw data before applying a threshold significantly improves the performance of the spike detector as spikes can be either negative or positive. The threshold is normally set as a scaled version of the root-meansquare or standard deviation of training data. According to [7], the threshold is defined as:

$$Thres\_data = 4\sigma_N \qquad \sigma_N = median\left\{\frac{|x(n)|}{0.6745}\right\} \tag{1}$$

where $x(n)$ is the training data at time n and σ_N is the estimated standard deviation of noise using N data points.

Recorded background noise originates from distant neuronal activities (i.e. spkies). These addictive noises can accumulate to a voltage level very close to the peak of the spikes. However, they are less spiky, i.e. contains lower frequency components (Fig. 2, top). To avoid wrongly identify these data points as spikes, NEO was proposed to filter out only high energy, high frequency data.

B. Nonlinear Energy Operator (NEO):

By emphasizing on both localized high energy and frequency of input signal, NEO is large only when signal is high in power (i.e. large magnitude) and changing fast (i.e. high frequency). Discrete time NEO is defined as:

$$\psi[x(n)] = x^2(n) - x(n-1) * x(n+1) \tag{2}$$

Fig. 2 illustrates how thresholds are applied to raw and NEO data. It is clear that NEO data provide a better contrast between spike and noise data. Outside the spikes, data are smoother. Therefore $x(n)$,$x(n-1)$ and $x(n+1)$ are similar in magnitude, leading to $\psi[x(n)]$ close to zero. This is true even when raw data is large in magnitude but not spiky. NEO is less prone to error even when noise accumulates to a high voltage level. The choice of NEO threshold value depends on data set and is normally obtained after a training period. In [5, 8], threshold value in NEO is defined as:

$$Thres\_NEO = 8\frac{1}{N}\sum_{1}^{N}\psi[x(n)] \tag{3}$$

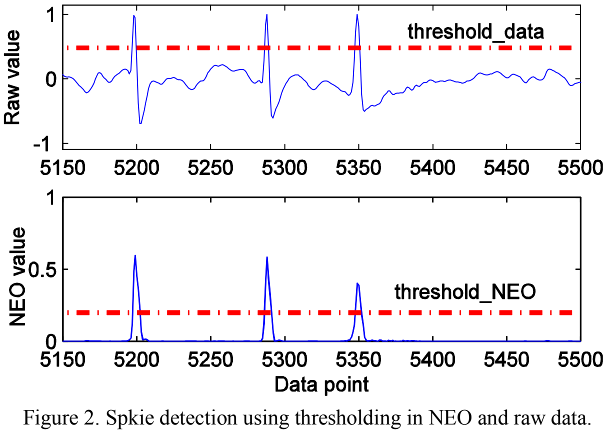

Figure 2. Spkie detection using thresholding in NEO and raw data.

III. Revised ROC Curve

NEO is generally accepted to be superior to Absolute Value. Area under the Receiver Operating Curves (ROC – Fig. 1) is normally used to compare the performance of each algorithm [8, 9]. Using this method, NEO is just slightly better than *Absolute Value* (0.926 vs 0.913). [8] pointed out that the difference is in fact statically better and more visually obvious when probability of detection (P_D) and probability of false alarm (P_{FA}) are plotted versus signal to noise ratio (SNR). In these works, P_D and P_{FA} are defined as:

$$P_D = 1 - \frac{No\ of\ misses}{No\ of\ true\ positve} \quad P_{FA} = \frac{No\ of\ false\ alarms}{No\ of\ true\ negatives} \tag{4}$$

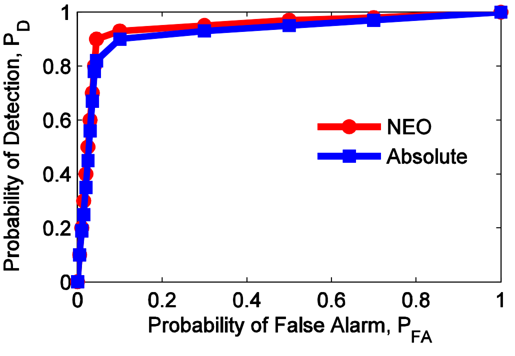

Figure 2. ROC curve for each detection method. The areas under the curves are used to measure their performance with larger area being better.

While these conclusions are valid, using P_D and P_{FA} to measure of these algorithms can easily be misleading. This is because neural recording data are very much skewed, i.e. there are much more noise data points than spike points. For example, using data available from [7], there are only about 3000 spikes among more than one million data points. As a result, 90% P_D translate to only 2700 spikes while 1% P_{FA} equals to almost 10000 points. Obviously we only interested in far left operating area of the ROCs in Fig. 2 as even a P_{FA} of 0.1 implies 100,000 false alarms which is more than 30× higher than the total true spikes available in the data set. Using number of true detections and number of false alarms, Fig. 3 confirms that these two algorithms become almost indistinguishable as the number of false alarms higher than 3000 points (sample 8 – i.e. a P_{FA} of 0.01) or even a few hundred (samples 1 to 7 - P_{FA} of less than 0.003). Thus the conventional ROC is not suitable to visually and quantitatively measure the performance of these algorithms in spike detection applications. Note that if ROCs are used in Fig. 3, performances of NEO and Absolute Value are almost indistinguishable in all 8 samples.

Instead of using probability of detection and false alarm, we can use the actual numbers. As a good detector should offer high detection while having low false alarm count, a performance indicator can be defined as:

$$P = \frac{No\ of\ true\ detection}{No\ of\ false\ alarm + 1} \tag{5}$$

One is added to the denominator to avoid dividing by zero in case of no false alarm. For each algorithm, a very low threshold value is initiated and swept to a very high value. At each threshold, P is recorded accordingly. Plotting P for each algorithm versus the iteration cycle would result in the curves shown in Fig. 4. Comparing Fig. 4 and Fig. 3; one can see that the new curves clearly differentiate the two algorithms with NEO has a much better performance.

The new *P-curve* can be interpreted as follows: Initially, threshold values are very small so both true spikes and a lot of negative results are captured. Since number of negatives is extremely large, P approaches zero for both cases. As the threshold increases, less false alarms are captured, resulting in a rising P-curve. These curves peak at a point where they capture the most spikes with least false negatives. As the iteration continues, higher threshold is applied, leading to less and less spikes detected. Thus P-curves starts to fall. When the threshold value becomes very large, no spike is captured and thus P is equal to zero.

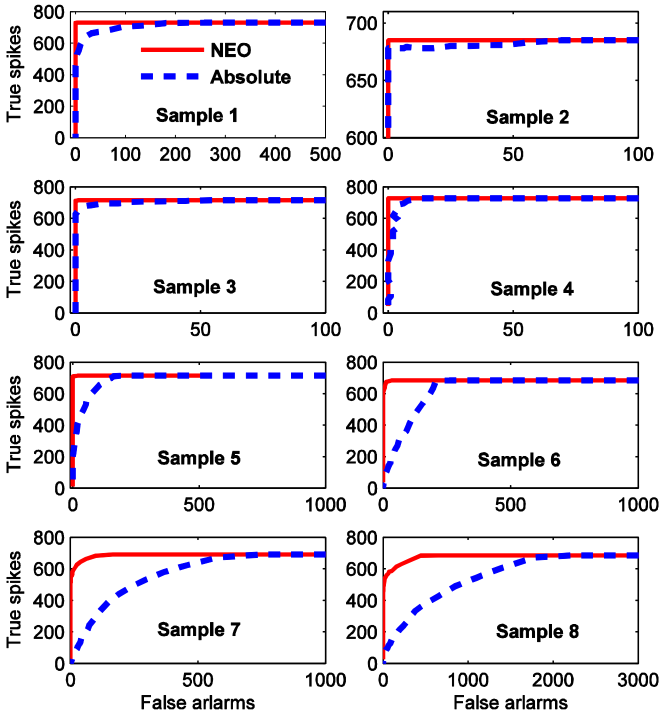

Figure 3. ROC comparison of NEO and Absolute Value using absolute value of true detection and false alarm.

Note that the area under the curve represents the probability choice, similar to the conventional ROC. The peak indicates the best operating point and thus a higher peak is more desirable. One can see in Fig. 4 that except for Sample 2, NEO significantly outperforms Absolute Value both in area under P-curve and peak value. Flat blue curves in Sample 6, 7, 8 show that Absolute Value performs very poorly as number of false alarms is always much higher than the number of identified spikes (although spikes can be detected). In Sample 2, Fig. 4, areas under the P-curves are almost the same, which agrees with that in Fig. 3 (note that all subplots in Fig. 3 are zoomed for a clearer view). However, the NEO curve rises first. This only implies good NEO threshold is available at an earlier iteration. Since they have similar areas and similar peaks, NEO and Absolute Value are equally good when used in Sample 2. Flat tops in the curves in Sample 2 suggest that as threshold increases, both true detections and false alarms do not change. It means that this sample is less noisy and there is a large margin window to choose threshold value.

The new curve is therefore more visually informative and easier to interpret. It is also useful as one can choose the best suitable threshold value based on the P-curve drawing.

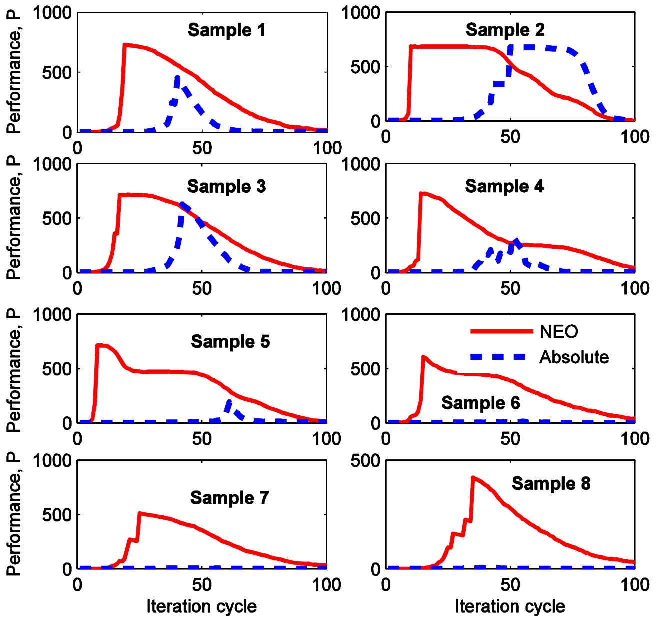

Figure 4. Proposed performance curve, *P-curve*, to evaluate NEO and Absolute Value on 8 data samples from [7]. In each sample, only first 300,000 data points are used. The new curve shows that NEO has a clear advantage over Absolute Value.

IV. Hybrid Absolute-NEO Spike Detector

A. Principle

Although NEO is superior when compared to Absolute Value, it requires more operations per cycles and thus power consumption. While Absolute Value only requires an 'absolute' operator, NEO requires two multipliers and a subtractor. Assuming that an absolute and a subtractor are similar to while a multiplier requires 10× operations of an adder, NEO is 20× more power hungry than Absolute Value.

We apply a threshold (*Thres_data*) to filter out most of low magnitude noises before passing the "potential spikes" to the NEO stage. Since majority of data points are of low magnitude, only a small portion can pass through the first stage. NEO is subsequently activated and calculates $\psi[x(n)]$. Fig. 5 shows the flow that of the proposed algorithm. If the number of potential spikes is low compared to the total data points, power consumption of the detector will be similar to the Absolute Value while its performance is close to that of NEO. The key is to choose a *Thres_data* that is conservative enough so that no true positive is missed. For example if we choose *Thres_data* equal to 3×rms(data) in the samples available in [7], 99% of data points are filtered out but some true spikes will be missed. As a result in "difficult sample" where both raw and NEO data are noisy, only 70% of true spikes are detected. On the other hand, lowering *Thres_data* to a very low value diminishes its power advantages. Fig. shows power improvement and normalized performance of the proposed algorithm when compared with the NEO when *Thres_data* equal to 2×rms(data). It is apparent that the new approach offers at least 8x power reduction (Fig. 6- top). Regarding the normalized performance, it detects the same number of true positives and false positives in 14 samples out of 20 samples. The worst case happens in sample 13 where the normalized performance is only 87.2%. This is because sample 13 is very noisy. Both Absolute Value and NEO originally perform very poorly in this sample as well.

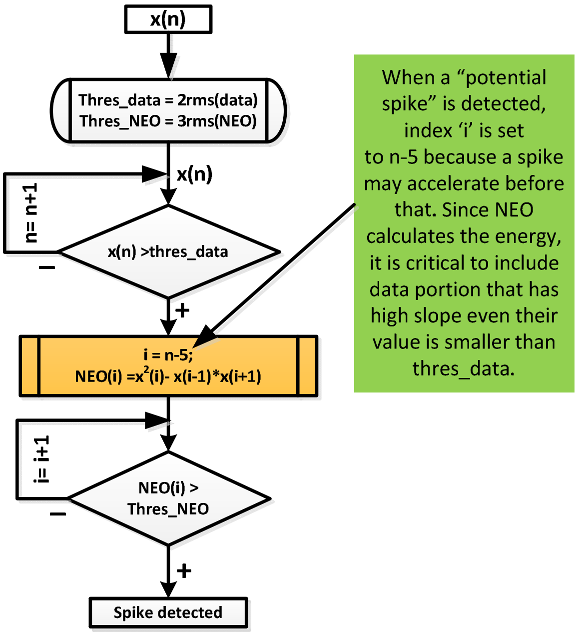

Figure 5. Flow chart of the proposed hybrid spike detector.

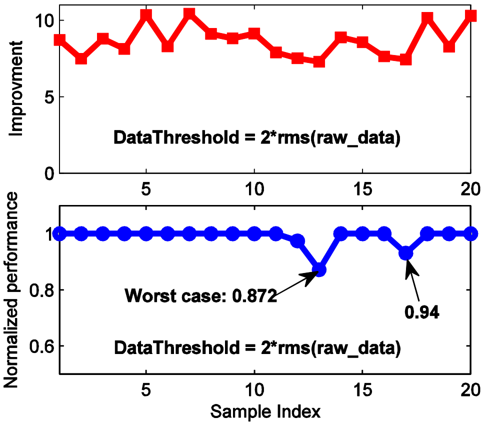

Figure 6. Normalized performance and power reduction of the proposed detector versus different value of scaling constant, C.

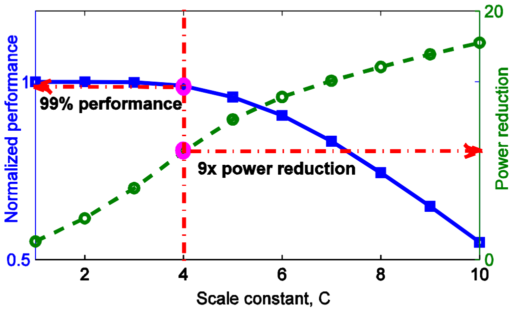

Figure 4. Normalized performance and power reduction of the proposed detector versus different value of scaling constant, C.Best suitable value of C is from 3 to 6. $Thres\_data = \frac{1}{2}C[rms[x(n)]$

We study the impact of the scaling factor, C, on the proposed method when compared to NEO. By sweeping C from 1 to 10, Matlab results are shown in Fig. 7. When C is small, performance of the hybrid method is slightly worse than NEO, even though their average performances are the same. As C increases from 3 to 6, it achieves a mean power improvement of 8 to 10 while ensure 99% performance of NEO, as highlighted in Fig. 7. With higher C however many spikes are missed, which is undesirable. We suggest a good scaling constant of 4 so that most spikes are detected with about 9× power reduction when compared to NEO method.

V. CONCLUSION

By combining Absolute Value and NEO, our hybrid detector offers similar performance as NEO while only consume one-ninth of power consumption. This architecture therefore is more suitable for implantable VLSI systems where stringent power consumption is the most critical constraint. In this paper, a *P-curve* is also proposed to evaluate the spike detector. The newly proposed curve is more intuitive and informative when compared to the conventional ROC. Performance of each detector, best threshold value as well as the characteristic of the input data can be inferred by plotting the P-curve versus the threshold.

REFERENCES

[1] A. M. Kamboh and A. J. Mason, "Computationally Efficient Neural Feature Extraction for Spike Sorting in Implantable High-Density Recording Systems," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on,* vol. 21, pp. 1-9, 2013.

[2] C. Moosung, L. Wentai, Z. Yang, C. Tungchien, K. Jungsuk, M. Sivaprakasam, *et al.*, "A 128-Channel 6mW Wireless Neural Recording IC with On-the-Fly Spike Sorting and UWB Tansmitter," in *Solid-State Circuits Conference, 2008. ISSCC 2008. Digest of Technical Papers. IEEE International*, 2008, pp. 146-603.

[3] L. Hongge, Y. Pan, and X. Tongsheng, "An efficient spike-sorting for implantable neural recording microsystem using hybrid neural network," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, 2012, pp. 5274-5277.

[4] C. Tung-Chien, M. Tsung-Chuan, C. Yun-Yu, and C. Liang-Gee, "Low power and high accuracy spike sorting microprocessor with on-line interpolation and re-alignment in 90nm CMOS process," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, 2012, pp. 4485-4488.

[5] S. Mukhopadhyay and G. C. Ray, "A new interpretation of nonlinear energy operator and its efficacy in spike detection," *Biomedical Engineering, IEEE Transactions on,* vol. 45, pp. 180-187, 1998.

[6] I. Obeid and P. D. Wolf, "Evaluation of spike-detection algorithms fora brain-machine interface application," *Biomedical Engineering, IEEE Transactions on,* vol. 51, pp. 905-911, 2004.

[7] R. Q. Quiroga, Z. Nadasdy, and Y. Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering," *Neural Computing,* vol. 16, pp. 1661-1687, 2004.

[8] S. Gibson, J. W. Judy, and D. Markovic, "Technology-Aware Algorithm Design for Neural Spike Detection, Feature Extraction, and Dimensionality Reduction," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on,* vol. 18, pp. 469-478, 2010.

[9] H. Linh, Z. Yang, and L. Wentai, "VLSI architecture of NEO spike detection with noise shaping filter and feature extraction using informative samples," in *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, 2009, pp. 978-981.