A Highly Accurate Spike Sorting Processor With Reconfigurable Embedded Frames for Unsupervised and Adaptive Analysis of Neural Signals

Majid Zamani, Dai Jiang, and Andreas Demosthenous
Department of Electronic and Electrical Engineering, University College London, London WC1E 7JE, UK
Email: {m.zamani, d.jiang, a.demosthenous}@ucl.ac.uk

***Abstract*—Future implantable devices demand ultra-low power consumption with self-calibration capability providing real-time processing of biomedical signals. This paper introduces an adaptive processing framework for highly accurate on-chip spike sorting processing by learning the signal model in the recorded neural data. The novel adaptive spike sorting processor employs dual thresholding detection, adaptive feature extraction and online clustering with sorting threshold self-tuning capability. A prototype chip was fabricated in 180 nm CMOS technology. It achieves 84.5% overall clustering accuracy, provides up to 240X data reduction and consumes 148 µW of power from a 1.8 V supply voltage.**

I. Introduction

Interactions between neurons are performed via electrical signals called action potentials or spikes. The information from spikes has led the development of brain machine interfaces (BMIs) which are capable of real-time processing at low power consumption. BMIs have been developed for therapeutic applications using the neural modulation of a particular pathway [1] or building a communication bridge to deliver lost motor commands to external assistive devices for patients with damaged sensory or motor functions [2]. The neuronal signals are recorded using implanted electrode arrays and the recorded neuronal activities are composed of multiple neurons. It is often desired to classify the recorded spikes to their source of origin for modeling application specific BMI set-ups. The process of mapping the recorded spikes to their source of origin is called *spike sorting*. It consists of four major steps: detection, alignment, feature extraction (*FE*), dimensionality reduction (*DR*) and clustering (see Fig. 1).

Different spike sorting processors have been developed for neuronal spike sorting [3]-[5]. In [3] the spike sorting processor utilizes template matching for clustering the monitored spikes. It is discussed in [6] that the clustering accuracy decreases dramatically when the similarity index between the spike templates are high. The spike template in [3] has 45 features ($K = 45$; K defines the feature space dimensionality), which requires both appreciable chip area for saving the transient clusters and power consumption for the training phase. It is desired to have less than ten features ($K < 10$) in implantable spike sorting processors to avoid memory issues in clustering.

An asynchronous spike sorting processor has been presented in [4], where asynchronous self-timed methodology suppresses the leakage power and standby power in the absence of variations. The crucial challenge in asynchronous methodology is designing complex circuits (e.g. clustering unit) which requires additional handshaking circuitry overhead. The spike sorting processor in [4] comprises detection, alignment and feature extraction units, and the clustering is performed in an external unit.

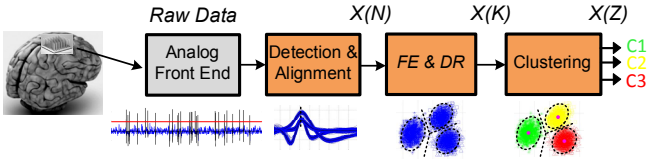

Fig. 1. Spike sorting chain for determining single unit activity. Data dimensionality is reduced over processing units ($Z < K < N$).

A real-time neural spike sorting processor has been described in [5] in which the feature extraction unit uses fixed coefficients to extract the spike features. This approach for retaining features is not efficient when the similarity level between the spike waveforms and the amount of noise vary. In [5] the clustering accuracy is 87% when the number of clusters are set manually and 72% when in online mode (which is less than the reported median clustering accuracy in [3]).

This paper presents the first generation of an adaptive spike sorting processor by further developing the work in [7]. The proposed adaptive processing framework offers integration of reconfigurable processing frames to complement the non-adaptive conventional synchronous spike processing systems (SSPSs). This proposed processor realizes an adaptive paradigm in which the functionality or structure of the processor tailors itself in the embedded frames by learning the characteristics of the input neural signals. As a result, unlike the SSPS architecture, which has been used in most of the integrated spike sorting processors due to its simple realization, the operation of the adaptive processor has robustness to the input signal variations [e.g. signal noise standard deviation (σ_N)].

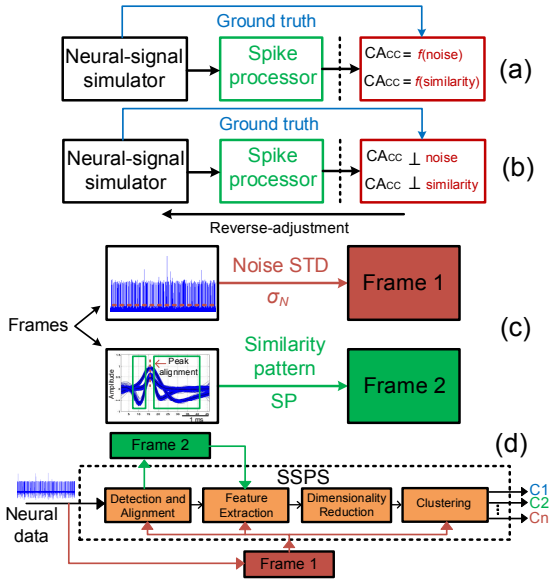

Fig. 2. (a) Traditional spike sorting in which the clustering accuracy (CAcc) is the function of noise [*f*(noise)] and similarity [*f*(similarity)]. (b) Illustration of spike processor independent (⊥) of recorded data noise and similarity of spike waveforms. In this class of processing it is expected that the CAcc is matched with ground truth and at any condition set in the neural-signal simulator. (c) Abstract view of mapping the proposed reverse-adjusted spike processor characteristics into the frames (Frame 1 and Frame 2) for implementing the adaptive concept. Frame 1 increases the processor noise robustness by sensing the noise standard deviation (STD) while Frame 2 adapts the similarity pattern (SP) between the aligned spike waveforms. (d) Transformation of SSPS based on the developed reconfigurable frames (Frame 1 and Frame 2) to an adaptive spike processor. The created frames are embedded to the main processing line. Frame 1 is embedded into the detection and alignment, feature extraction and clustering units. Frame 2 is embedded into the feature extraction unit.

II. Development of Reconfigurable Embedding Frames for Spike Sorting

This section discusses the motivation behind the selection of embedding frames for spike sorting to maintain optimal clustering performance under any conditions. The processing outcome is then essentially independent from the input signal. Two key factors in spike sorting performance degradation are the noise of recorded data and the similarity index between the spike waveforms. The aim is to develop a spike processor in which the performance is adjusted to an optimal level (maintaining lowest clustering error) with the varying difficulty between the recorded spike waveforms in recording channels and different noise levels.

The concept of the frame modeling procedure is shown in Fig. 2. Fig. 2(a) is the block diagram of a traditional spike processor (SSPS) whose performance varies as a function of recording channel noise [*f*(noise)] and similarity of extracted spike waveforms [*f*(similarity)]. Fig. 2(b) shows the spike sorting concept developed with added reverse-adjustment flow. In this class of spike sorting processing, it is required that the resulting clustering performance (CA_{CC}) is essentially independent (⊥) of noise and spike shape similarity between the extracted spikes. The reconfigurable frames are developed based on the reverse-adjustment concept in spike sorting as shown in Fig.

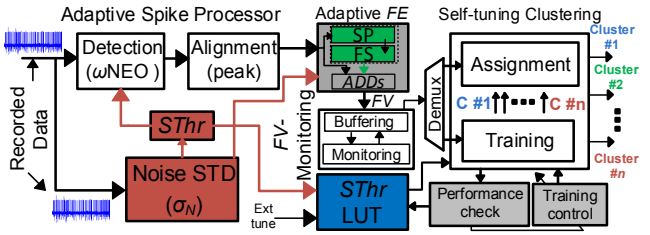

Fig. 3. Adaptive spike sorting block diagram. The developed frames in Fig. 2 (Frame 1 and Frame 2) are embedded in the spike processor as Frame 1 (= σ_N) and Frame 2 (= SP). n = 1,…,6 represents the number of existing clusters (Cluster #1 … Cluster #n) in the recorded data.

2(c). Adding the created frames (Frame 1 and Frame 2) to the traditional spike processor presents a new approach for mapping the recorded spikes to the individual neurons. The created frames provide the functions of *adaptivity* or *reconfigurability* by distinguishing the noise property of the neural data stream (Frame 1) and the similarity between active neurons (Frame 2) to maintain the clustering performance at an optimal level. Fig. 2(d) shows the two frames embedded in the spike sorting system described in [7]. The adaptive processing provides an on-chip tuning mechanism for programming the key coefficients in the relevant building blocks.

The Frame 1 and Frame 2 are translated into mathematical models in the appropriate circuit blocks. Frame 1 provides noise robustness to the processing chain by modeling the noise standard deviation (STD) of the recorded neural signal (σ_N) which is calculated by a median processor. Frame 2 provides similarity robustness by modeling of the localized differences extraction of the aligned spikes introduced in [8].

III. Functional Description of Adaptive Spike Sorting Processor

This section outlines the system level implementation of an adaptive spike processor based on the concept in Fig. 2. Fig. 3 shows the adaptive spike sorting processor. The developed frames are embedded into the sorting system. Frame 1 monitors the noise STD (σ_N) and defines the sorting threshold ($SThr = 4\sigma_N$) [9] which is distributed to the detection block, adaptive *FE* and the *SThr* look-up-table (LUT). Frame 2 is developed to extract the similarity pattern (SP) between the aligned spike waveforms in the recording channel [8]. The SP is sent to the frequency synthesizer (FS) used in the adaptive *FE* to examine the slope variations of the extracted local differences (amount of dilation/contraction) for tuning the decomposition lines to the sub-bands with most informative features in the adaptive discrete derivatives (*ADDs*) unit. Frame 2 in conjunction with FS provides similarity robustness to the spike sorting chain. The feature vectors (*FVs*) are sent to the *FV*-monitoring unit and subsequently to the clustering unit for training and assignment phases.

To obtain high detection accuracy while keeping the power consumption low, a complementary approach is used in which *SThr* is considered as a conditional activation function of the modified version of nonlinear energy operator [10], termed as

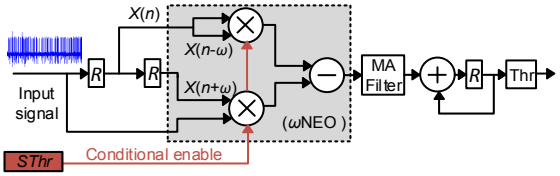

Fig. 4. ωNEO conditional control. Conditional enable is initiated by $SThr = 4\sigma_N$). Moving-average (MA) filter is applied to signal energy for reducing the effect of projected noise in detection threshold (Thr) calculation. $\omega = 2$ in this design.

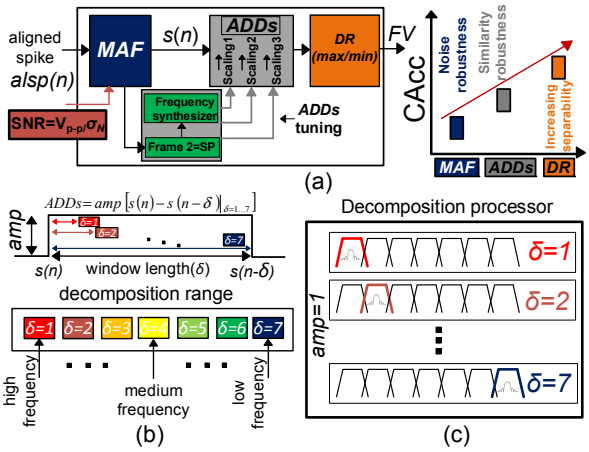

Fig. 5. (a) Adaptive *FE* unit. The *MAF* suppresses the effect of random and high frequency noise of an aligned spike waveform [*alsp(n)*]. Frame 2 (= SP), FS and *ADDs* provide the adaptive decomposition (SP →FS → *ADDs*). Three decomposition lines are selected based on (scaling1, scaling2 and scaling3) for multi-resolution decomposition. The *DR* block reduces dimensionality by extrema (max/min) sampling of decomposed spike waveform *s*(*n*). (b) Demonstration of *FE* processor employing spectral analysis in *ADDs* (*amp* = 1). Decomposition intensity range is shown with different colors from high ($\delta = 1$) to low ($\delta = 7$). (c) Illustration of *ADDs* as adaptive filtering.

ωNEO. This method has two advantages: 1) conditional enabling is directly applied to the ωNEO which is composed of two multipliers and one subtractor as shown in Fig. 4. Thus, when the input exceeds *SThr*, the true identity of the spikes is examined using ωNEO which provides a double check on accuracy (*SThr*-ωNEO); and 2) power saving due to dual-thresholding results in 30% power reduction based on Cadence synthesis simulation.

The architecture of the adaptive *FE* unit is shown in Fig. 5. It consists of four main sub-units namely a moving average filtering (MAF), frequency synthesizer (FS), adaptive discrete derivatives (*ADDs*) and dimensionality reduction (*DR*) unit. The *MAF* acts as a denoising filter to improve the *FE* robustness to random noise (out-of-band noise) while retaining the crucial encoded information buried in spikes. The *MAF* length is adjusted by the SNR ratio $V_{p\text{-}p}/\sigma_N$ ($V_{p\text{-}p}$ is the peak-to-peak voltage value of the recorded neural data and σ_N is provided in Frame 1). The next block of the adaptive *FE*, are the *ADDs* which calculate the slope at each sample point over a number of different time scales:

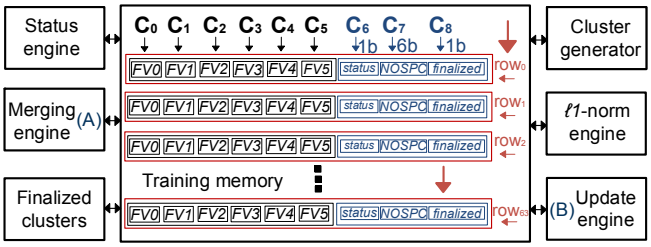

Fig. 6. Illustration of training unit structure which is composed of training memory and peripheral processing engines. Each row of training memory consists of six columns (C_0-C_5) for accommodating extracted features (*FV0*-*FV5*), a 1-bit status flag C_6 for dynamic power saving, 6 bits for number of spikes per cluster (NOSPC) in C_7 for cluster mean update in (A); C_7 is also used for cluster generation and checking the finalized cluster means in training phase; and a 1-bit finalized flag C_8 for conditional initiation of (A) and (B). The chosen number of interleaving processing in $\ell 1$-norm and merging engines is 8 to minimize the power-area product.

$$ADDs = amp\left[s(n) - s(n-\delta)\big|_{\delta=1\ldots7}\right] \quad (1)$$

where *amp* = 1) is the amplitude of the decomposition window, *s* is the spike waveform, *n* is the sample point and δ is the scaling factor (time delay). The equation shows subtraction between the samples *n* and $n-\delta$. Multi-resolution decomposition of a spike can be obtained if the scaling factor δ is swept over a wide range as demonstrated in Fig. 5(b)-(c). In *ADDs*, the decomposition window length δ is not fixed and will be tuned based on the output of FS intermittently. Adjustment of the scaling factors (scaling1, scaling2 and scaling3) is based on the three frequency sub-bands from $\delta = 1$ to $\delta = 7$ which correspond to the most informative features for clustering.

The algorithm proposed in [11] is used for real-time and unsupervised clustering of neurons. As shown in Fig. 3, performance check and training control units are exploited in the clustering unit to enhance the clustering median accuracy by incorporating a sorting threshold self-tuning scheme. The performance check unit monitors and evaluates the clustered *FVs* based on the defined performance metrics. It decides whether the level of the sorting threshold should be iteratively adjusted to an optimal level (T_{opt}) [7] and triggers retraining to re-compute the cluster means if needed. The block diagram of the training memory structure and training unit main processing engines (e.g. status engine) are shown in Fig. 6. The transient memory core is implemented in a matrix format to have access to the memory locations.

IV. Measured Results

The proof-of-concept adaptive spike sorting processor was fabricated in a 180 nm CMOS technology. The die microphotograph is shown in Fig. 7. The chip occupies an area of 10 mm^2. The processor uses four different clock rates (30 kHz, 120 kHz, 240 kHz, 960 kHz) to obtain the best processing efficiency which results in 148 µW of power from a 1.8 V supply voltage. A standard neural database [12] with a known ground truth (spike times and classes) was used to

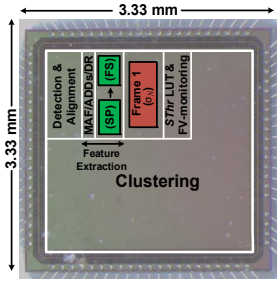

Fig. 7. Die photo of the adaptive spike sorting processor chip.

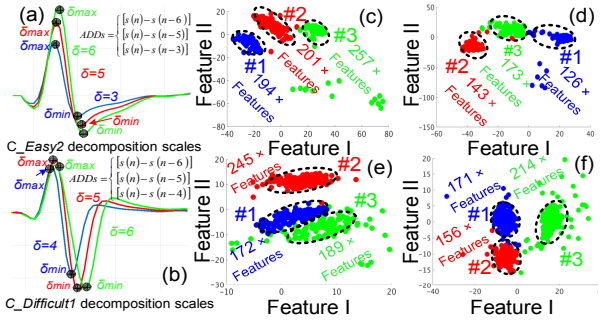

Fig. 8. An illustration of extrema features using different sets of scaling factors. The scaling factors are selected based on the frequency synthesizer (FS) output for (a) C_*Easy2_0.05* and (b) *C_Difficult1_0.05*. 2-D projection of clusters for (c) *C_Easy1_0.05*, (d) C_*Easy2_0.05,* (e) *C_Difficult1_0.05* and (f) *C_Difficult2_0.05*. (Spikes have been colored according to the ground truth). The number of features allocated to each cluster in assignment phase for the duration of 7 seconds.

validate the functionality of the fabricated processor. Fig. 8(a)-(b) shows the specific cases in which different scaling factors are used for decomposition of spike waveforms. Using the identified scaling factors in *ADDs* introduces more discrimination for clustering. The clustering accuracy of the adaptive spike sorting chip was tested and evaluated across all datasets and noise levels. In the *DR* unit, extrema sampling of decomposed spike waveforms is performed to retain six features ($K = 6$) for clustering. Fig. 8, from (c) to (f) shows the two-dimensional (2-D) projection of the clusters in different datasets form Easy1 to Difficult2 [12]. The boundaries of the clusters are marked with dotted lines. The overall clustering accuracy of 84.5% is obtained over all different datasets and noise levels. Table I provides a comparison with other spike processors. The processor in this paper is the first adaptive spike sorting processor which provides on-chip parametric tunability with the developed frames (Frame 1 and Frame 2). Compared to the spike processor in [3], it achieves almost 10% higher online clustering accuracy (CA_{CC}). The power density of the adaptive spike sorting processor (54.8 µW/mm^2) is lower than the processors in [3] and [5].

V. Conclusion

An adaptive processing framework has been proposed to maximize the clustering performance by learning the neural signal statistics in the embedded reconfigurable frames (Frame 1 and Frame 2), and to minimize the implementation cost using hardware efficient resources to. As proof of concept, an adaptive spike sorting processor has been designed, fabricated and evaluated using standard neural datasets. In the presence of input neural signal variations its 84.5% overall clustering accuracy outperforms the state-of-the-art.

TABLE I: Comparison with Previous Work

Reference	[3]	[5]	This Work
Detection	✓	✓	✓ (***SThr-ω*NEO**)
Alignment	✓	✓	✓ **(Peak)**
Feature extraction	x	✓	✓ **(Adaptive)**
Clustering	✓	✓	✓ **(Self-tuning - T_{opt})**
Compression factor	240X	257X	150X[(b)] / 240X
Power (µW/channel)	4.68	0.175	148[(c)]
Area (mm^2/channel)	0.07	0.003	2.7[(d)]
Power density (µW/mm^2)	66.8	58.33	54.8
Process (nm)	65	65	180
Core voltage (V)	0.27	0.54	1.8
Clustering accuracy	75%	72-87%[(a)]	84.5%
Adaptive design	x	x	✓ **(Frame 1 & Frame 2)**

(a) 87% average accuracy when the number of clusters are set manually.
(b) Compression factor in error monitoring mode.
(c) The synthesized processor power is 20 µW from 1.1 V in 45 nm NAN-GATE.
(d) Power density is calculated based on assignment unit area in Fig. 3.

References

[1] A. Mohammed, M. Zamani, R. Bayford, and A. Demosthenous, "Patient specific Parkinson's disease detection for adaptive deep brain stimulation," *EMBC 2015*, pp. 1528–1531, Aug. 2015.

[2] M. Velliste, S. Perel, M. C. Spalding, A. S. Whitford, and A. B. Schwartz, "Cortical control of a prosthetic arm for self-feeding," *Nature*, vol. 453, pp. 1098–101, 2008.

[3] V. Karkare, S. Gibson, and D. Markovic´, "A 75-µW, 16-channel neural spike-sorting processor with unsupervised clustering," *IEEE J. Solid-State Circuits*, vol. 48, pp. 2230–2238, Sep. 2013.

[4] T.-T. Liu and J. M. Rabaey, "A 0.25 V 460 nW asynchronous neural signal processor with inherent leakage suppression," *IEEE J. Solid State Circuits*, vol. 48, no. 4, pp. 897–906, Apr. 2013.

[5] S. M. A. Zeinolabedin, A. T. Do, D. Jeon, D. Sylvester, and T. T. Kim, "A 128-channel spike sorting processor featuring 0.175 µW and 0.0033 mm^2 per channel in 65-nm CMOS," *Symp. VLSI Circuits*, Jun. 2016.

[6] Y. Yuan, C. Yang, and J. Si, "The m-sorter: an automatic and robust spike detection and classification system," *J. Neurosci. Methods,* vol. 210, pp. 281–290, Sep. 2012.

[7] M. Zamani and A. Demosthenous, "Feature extraction using extrema sampling of discrete derivatives for spike sorting in implantable upper-limb neural prostheses," *IEEE Trans Neural Syst. Rehabil. Eng.*, vol. 22, pp. 716–726, Jul. 2014.

[8] M. Zamani, and A. Demosthenous. "Dimensionality reduction using asynchronous sampling of first derivative features for real-time and computationally efficient neural spike sorting," *ICECS 2013*, pp. 237–240, Dec. 2013.

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

[10] S. Mukhopadhyay and G. Ray, "A new interpretation of nonlinear en-ergy operator and its efficacy in spike detection," *IEEE Trans. Biomed. Eng.*, vol. 45, pp. 180–187, Feb. 1998.

[11] U. Rutishauser, E. M. Schuman, and A. N. Mamelak, "Online detection and sorting of extracellularly recorded action potentials in human medial temporal lobe recordings, in vivo," *J. Neurosci. Methods,* vol. 154, pp. 204–224, Jun. 2006.

[12] [Online] http://www2.le.ac.uk/centres/csn/research-2/spike-sorting