Proceedings of the 2nd International IEEE EMBS
Conference on Neural Engineering
Arlington, Virginia · March 16 - 19, 2005

Low–Power Architectures for Spike Sorting

Alex Zviagintsev, Yevgeny Perelman and Ran Ginosar
VLSI Systems Research Center, Technion—Israel Institute of Technology, Haifa 3200, Israel
[alehan@tx.technion.ac.il]

***Abstract*–Front-end integrated circuits for spike sorting will be useful in neuronal recording systems that engage a large number of electrodes. Detecting, sorting and encoding spike data at the front-end will reduce the data bandwidth and enable wireless communication. Without such data reduction, large data volumes need to be transferred to a host computer and typically heavy cables are required which constrain the patient or test animal. Front-end processing circuits must dissipate only a limited amount of power, due to supply constraints and heat restrictions. Two reduced complexity spike sorting algorithms are introduced, one based on Integral Transform and another on segmented PCA. The former achieves 98% of the precision of a PCA sorter, while requiring only 2.5% of the computational complexity. The latter algorithm is somewhat more accurate but incurs a higher complexity.**

I. INTRODUCTION

Various methods have been proposed for the separation and sorting of action potential waveforms ("spikes") originating from different neurons [1]-[11]. Spike sorting requires high bandwidth communications between the electrodes and the sorting computer, as well as high-performance processing. When a large number of signals are to be handled, typical transmission resources are insufficient [12]. We investigate *Neuroprocessors*, which are front-end integrated circuits for processing neuronal recordings, with spike sorting capabilities [13]. In this paper we focus on spike sorting algorithms and architectures that trade off some classification accuracy in return for significant savings in power. The power requirements of hardware spike-sorting have also been discussed in [14].

We introduce the *Integral Transform* (IT) and the *Segmented PC* algorithms for spike sorting, and compare them with an algorithm based on principal components analysis (*PCA*) [1]. The IT algorithm achieves 98% classification accuracy for about 2.5% of the computational effort of PCA spike sorting. The Segmented PC algorithm, which combines features of both IT and PCA, is more accurate than the IT algorithm by a fraction of one percent, but incurs twice or three times the complexity. The Neuroprocessor system is overviewed in Section II. The algorithms and architectures are described in Section III, and their performance is analyzed in Section IV.

II. SYSTEM OVERVIEW

In a typical experimental setup, the signals recorded by the electrodes are amplified and transmitted over wires to a host computer where they are digitized and processed according to experimental requirements [15]. The main disadvantage of this arrangement is the need to connect a cable to the subject, restricting its movement. Instead, the Neuroprocessor chip performs front-end data processing [13] and data reduction by means of spike detection and sorting to enable bidirectional wireless communications that replace cables and allow free movement of the patient or test subject. While it is feasible to transfer some raw signal recordings over the wireless channel, combining data collected by a large number of electrodes is prohibitive [12], and additional data reduction must be carried out by the Neuroprocessor. In many neuronal experiments, the most important data is the indication of spikes, their sources (electrode and identifiable unit within the electrode), and the time of their occurrence. These indications are produced with a real time spike sorting algorithm, and the Neuroprocessor transmits only spike indications and avoids sending the raw signal. Communicating such indications requires much lower bandwidth and could be made feasible with low-power wireless links.

The raw signal from a neuronal recording electrode may be sampled, for instance, at 25 Ksps and digitized at 12 bits/sample, producing a data rate of 300Kbits/second. The Neuroprocessor generates a 20-bit *spike notification message* including a time stamp and electrode and unit identification. Assuming a (relatively high) spike rate of 100 spikes/second per electrode, the expected data rate is 2Kbits/second, less than 1% of the raw data rate. The architecture of the part of the Neuroprocessor that processes the signal from a single electrode is shown in Figure 1. Spike detection and sorting must be adaptable, due to unstable recording conditions [5]. Therefore, raw data is transmitted to the host computer periodically for training, and recalculated parameters are sent back to the Neuroprocessor. At all other times, the spike signal is processed by the Detector and Sorter. The Detector identifies the presence of spikes in the input, determines their starting point, and initiates the operation of the Sorter. In this paper we focus on sorting algorithms and assume a given Detector. The output logic produces the spike notification message.

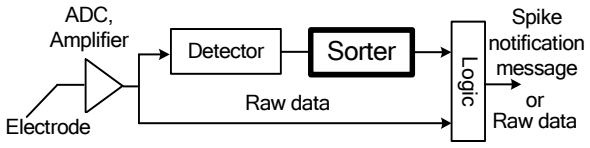

Figure 1: Neuroprocessor Architecture: one electrode section

III. SPIKE SORTING ALGORITHMS AND ARCHITECTURES

When exploring algorithms and VLSI architectures for real time spike sorting to be carried out at the Neuroprocessor, we seek to minimize power dissipation while still achieving acceptable levels of sorting accuracy. Following [14], we consider the relative computational complexity of different architectures as a predictor of their power requirements.

Signal processing VLSI architectures may employ analog computations to reduce power dissipation [16]. In the following, we note the cases where we may benefit from analog computations, but the comparative analysis of the architectures is carried out in the digital domain.

A. The Integral Transform Algorithm

The *Integral Transform* (IT) sorting algorithm classifies the spikes projected onto the two-dimensional Integral Transform space. We assume that the spike signal can be split into two time intervals *A* and *B*, according to its positive and negative phases (Figure 2). Therefore, the two axes of the IT space represent the normalized discrete signal integrals over *A* and *B* (Figure 3):

$$I_A = \frac{1}{N_A}\sum_{i=1}^{N_A} x(i) \quad , \quad I_B = \frac{1}{N_B}\sum_{i=1}^{N_B} x(i) \quad .$$

Spike classification is based on linear separation in IT space, as follows:

$$I_B > m \cdot I_A + n \;\Rightarrow\; set\ I, \quad otherwise \;\Rightarrow\; set\ II$$

Here $m, n \in \Re$ are the parameters of the separation line (Figure 3). They are determined by off-line learning, which may be based on any appropriate technique, such as SVM [17].

Linear classification has been selected in an attempt to minimize hardware and computational complexities. In simple cases, one line may suffice for sorting spikes into two clusters. In general, any number of lines may be employed, either to further constrain the classification space, or to enable sorting into three or more clusters, or both.

A conceptual VLSI architecture for the IT algorithm is shown in Figure 4. The input is integrated over the first time interval (A) and the result is stored as I_A (storage is omitted from the figure). During the second interval (B), the integrator generates I_B. Subsequently, the parameters *m* and *n* are used to generate mI_A+n that is compared to I_B. A mixed-signal version of this architecture could employ an analog integrator at the input, resulting in reduced rate analog-to-digital conversion.

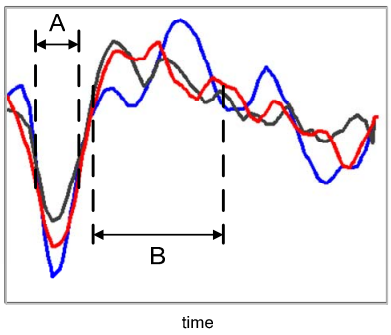

Figure 2: Spike representation by two integration time intervals *A, B*

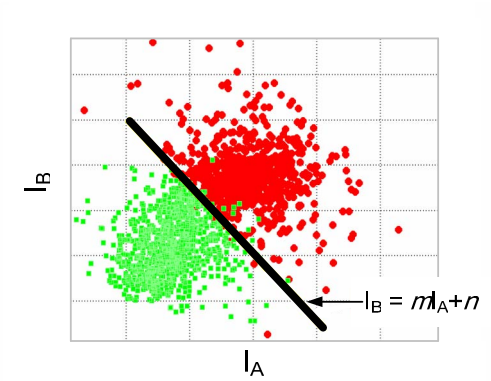

Figure 3: Spikes projection on the Integral Transform (IT) space

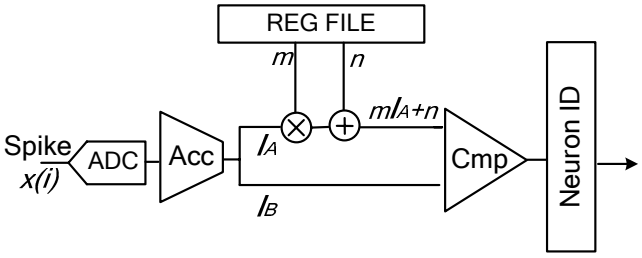

Figure 4: IT VLSI architecture

B. Principal Component Sorting

A conceptual VLSI architecture for on-chip sorting by means of principal component analysis (PCA) using two principal components and linear classification is shown in Figure 5. Each input sample is multiplied by two PC coefficients, and the two accumulated projections are linearly compared.

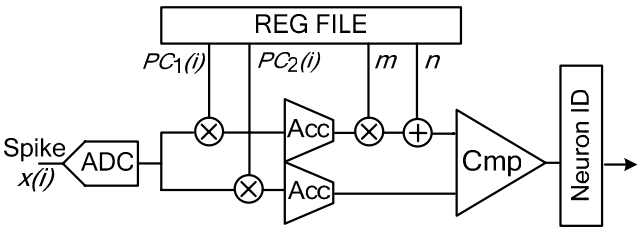

Figure 5: PC VLSI architecture

C. Segmented Principal Component Sorting

The segmented PC algorithm on k segments (kPC) approximates PCA while using a reduced number of multiplications. The signal is integrated over several time intervals (k_1 segments for PC_1 and k_2 segments for PC_2). Each integral is multiplied by the average of the principal component values over the same interval, as follows:

$$S_1 = \sum_{p=1}^{k_1} I_1(p) \cdot \alpha(p)$$

where

$$I_1(p) = \sum_{i \in \text{segment } p} x(i)$$

and

$$\alpha(p) = \underset{i \in \text{segment } p}{Average} \{ PC_1(i) \}.$$

The expressions for PC_2 are similar. A VLSI architecture for the kPC algorithm is shown in Figure 6.

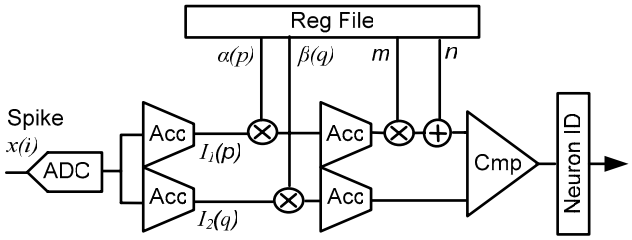

Figure 6: Segmented PC VLSI architecture

Another level of savings in computational complexity can be achieved by approximating the multiplication coefficients $\alpha(p)$ and $\beta(q)$ by powers of 2 (thus, multiplications are achieved by simple bit-shifting). This reduced complexity version of kPC is designated kPC$_2$ in the next section.

IV. Results

A. Algorithm Validation

The hardware spike sorting algorithms described above are compared (by simulation) to a software implementation of PCA [18]. Details of the PCA sorting are described in [19]. All algorithms are applied to the same data set. Figure 7 illustrates the algorithm validation scheme. First, part of the data is used for training, producing configuration parameters for the hardware algorithm. Second, the parameters are downloaded to the Neuroprocessor simulation model. Third, a simulation of the Neuroprocessor spike-sorting algorithm is applied to the entire data set, and the results are compared with the output of the software (PCA) algorithm.

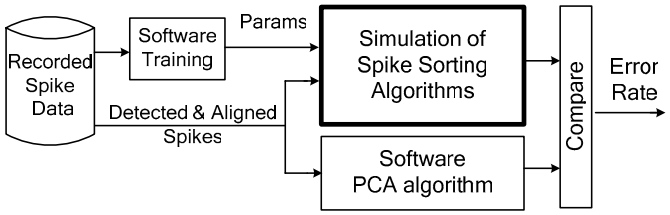

Figure 7: Algorithm validation

B. Spike Recording Method

Real spike data was taken from electrophysiological recordings of multiple spike trains, obtained from microelectrodes implanted in multiple cortical regions [20]. Neuronal signals from the electrodes were amplified, bandpass filtered (300 – 6000 Hz, four poles Butterworth filter), and sampled at 24 Ksps/electrode. A software algorithm for spike detection was first applied; only stable spike trains (as judged by stable spike waveforms, stable firing rate and consistent responses to behavioral events) were included in this study. The data set contains about 1,000 spikes per cluster. The spike sorting algorithms have been applied to this data under a number of simplifying assumptions, ignoring classification errors such as overlapping signals, burst-firing neurons and non-stationary background noise.

C. Analysis

The reduced computational complexity of the proposed VLSI algorithms comes at the expense of precision. We count the number of additions and multiplications required in each case per each spike. Multiplication is counted as about ten additions, and computational complexity is expressed as the total number of equivalent additions. The computational complexity is roughly proportional to the relative power consumption of each algorithm [14]. The results are shown in Table 1. The two versions of the kPC algorithm are tested with seven segments per each component. It is evident that the IT spike sorting algorithm achieves about 97.8% accuracy at about 2.5% of the complexity of the full PCA. The very small added accuracy levels offered by kPC and kPC$_2$ come at the price of 2-3 times higher complexity.

TABLE 1: COMPUTATIONAL COMPLEXITY AND CLASSIFICATION ERRORS OF THE SPIKE SORTING ARCHITECTURES

Algorithm	Add	Mult	Computational Complexity	Error Rate
PCA	400	400	4400	0.0%
7PC	165	15	315	1.4%
7PC$_2$	175	1	185	1.7%
IT	100	1	110	2.2%

V. CONCLUSIONS

We have described low-power architectures for spike sorting integrated circuits. Such systems may be implanted near recording electrodes, or employed as front end electronics for large multi-electrode arrays in either research or clinical applications. These systems enable substantial reduction of the communication bandwidth, which is essential when a large number of recording electrodes is involved.

Three VLSI architectures for spike sorting were compared: IT integrates the signal in segments, PC implements the common PCA analysis and segmented PC combines features of both. All three algorithms employ linear classification in the appropriate two-dimensional spaces.

The algorithms have been simulated with real neuronal spike data. The results are analyzed in terms of classification errors (relative to sorting results achieved with software PCA classification). The computational complexity of each algorithm was estimated based on the number of additions and multiplications involved.

The IT algorithm yields only marginal accuracy degradation relative to full PCA, while incurring only a small fraction of the computational complexity. Consequently, it is proposed for power-efficient spike sorting in neuronal processing integrated circuits.

ACKNOWLEDGMENT

This research was funded in part by a grant from the Office of the Chief Scientist, Israel Ministry of Industry and Trade.

REFERENCES

[1] Abeles M, Goldstein M.J., "Multispike train analysis," *IEEE Trans. Biomed. Eng*., 65:762–73, 1977.

[2] D'Hollander E.H. and Orban G., "A Spike recognition and on-line classification by an unsupervised learning system," *IEEE Trans. Biomed. Eng.* 26:279–84, 1979.

[3] Worgotter F., Daunicht W.J. and Eckmiller R., "An on-line spike form discriminator for extracellular recordings based on an analog correlation technique," *J. Neurosci. Methods*, 17:141-151, 1986.

[4] Jansen R.F., "The reconstruction of individual spike trains from extracellular multineuron recording using a neural network emulation program," *J. Neurosci. Methods*, 35: 203-213, 1990.

[5] Chandra R. and Optican L.M., "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.*, 44: 403–412, May 1997.

[6] Fee M.S., Mitra P.P. and Kleinfeld D., "Automatic sorting of multiple-unit neuronal signals in the presence of anisotropic and non-gaussian variability," *J. Neurosci. Methods,* 69: 175–188, 1996.

[7] Lewicki, M.S., "A review of methods for spike sorting: the detection and classification of neural action potentials." Network: Comp. Neural Syst. 9(4): 53-78, 1998.

[8] Mussa-Ivaldi F.A., Miller L.E., "Brain–machine interfaces: computational demands and clinical needs meet basic neuroscience," *Trends Neurosc.,* 26: 6, 329-334, 2003.

[9] Sahani M.M., "Variable models for neural data analysis," Ph.D. dissertation, California Inst. Technol., Pasadena, CA, 1999.

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

[11] Letelier J, Wever P. "Spike sorting based on discrete wavelet transform", J Neurosci Methods;101:93_106. 2000.

[12] Harrison R., "A low-power integrated circuit for adaptive detection of action potentials in noisy signals," Proc. 2003 Intl.Conference of the IEEE EMBS, Sept. 17-21, 2003.

[13] Perelman Y., Ginosar R., "Analog Front-end IC for Multi-channel Neuronal Recording System with Spike and LFP Separation," CCIT Tech. Rep. 513, Elec. Eng. Dept., Technion, 2004 (www.ee.technion.ac.il/~ran).

[14] Zumsteg Z.S., Ahmed R.E., Santhanam G., Shenoy K.V., Meng T.H., "Power Feasibility of Implantable Digital Spike-Sorting Circuits for Neural Prosthetic Systems," Proc. 26th Ann. Int. Conf. IEEE EMBS, pp. 4237-4240, Sept. 2004.

[15] Nicolelis M., "Actions from thoughts," *Nature*, 409:403-407, 2001.

[16] Mead C.," *Analog VLSI and Neural Systems,* Addison-Wesley, 1989.

[17] Vapnik V.N., *Statistical Learning Theory*, John Wiley & Sons, 1998.

[18] Alpha-Omega Engineering Co., Ltd, Alpha Sort Ref. Manual, 2002.

[19] Bar-Gad I., Ritov Y., Vaadia E., and Bergman H., "Failure in identification of multiple neuron activity causes artificial correlations," *J. Neurosci. Methods*., 107:1-13, 2001.

[20] Eytan D., Brenner N. and Marom S., "Selective Adaptation in Networks of Cortical Neurons, " *J. Neurosci. Methods,* 23(28):9349-9356, 2003