A New Method for Neural Spike Alignment: The Centroid Filter

Benjamin Metcalfe, *Member, IEEE,* Chris Clarke, *Member, IEEE,* Nick Donaldson, and John Taylor

***Abstract*—Recordings made directly from the nervous system are a key tool in experimental electrophysiology and the development of bioelectronic medicines. Analysis of these recordings involves the identification of signals from individual neurons, a process known as *spike sorting*. A critical and limiting feature of spike sorting is the need to align individual spikes in time. However, electrophysiological recordings are made in extremely noisy environments that seriously limit the performance of the spike-alignment process. We present a new *centroid*-based method and demonstrate its effectiveness using deterministic models of nerve signals. We show that spike alignment in the presence of noise is possible with a 30 dB reduction in minimum SNR compared to conventional methods. We present a mathematical analysis of the centroid method, characterising its fundamental operation and performance. Further, we show that the centroid method lends itself particularly well to hardware realisation and we present results from a low-power implementation that operates on an FPGA, consuming 10 times less power than conventional techniques - an important property for implanted devices. Our centroid method enables the accurate alignment of spikes in sub-0 dB SNR recordings and has the potential to enable the analysis of spikes in a wider range of environments than has been previously possible. Our method thus has the potential to influence significantly the design of electrophysiological recording systems in the future.**

***Index Terms*—action potentials, centroid, electrophysiology, spike alignment, spike sorting.**

I. Introduction

EXTRACELLULAR recording of physiological neural activity is the dominant experimental technique in electrophysiology providing essential information about both the central and peripheral nervous systems [1]. Extracellular electrodes such as hooks or suction electrodes are considered to be both reliable and minimally invasive recording approaches. These types of electrodes spontaneously record the electrical activity from an unknown number of excited axons that may serve different functions such as movement or memory. The study of these high-order functions often requires simultaneous recording from large areas of interconnected neurons using, for example, multi-channel electrode arrays [2]. There is a special interest in reconstructing the waveform of individual neurons from these multi-channel recordings so that the firing rates, and thus the encoded information, can be extracted and compared. This procedure is referred to as neural *spike sorting*, which is the labelling and classification of individual *action potentials* (APs) based on both shape and amplitude. It is generally considered that the shape and amplitude of APs recorded from a single neuron are time invariant and are a function of the axon diameter and the distance from the neuron to the recording electrode.

Historically, the spike sorting process was performed visually by a researcher who, after first determining the number of different classes, would classify each AP based on shape and reconstruct individual spike trains for each neuron. This time-consuming process has since been replaced by more reliable computer based automated methods that make use of statistical analysis and signal processing [3]–[5]. There are many different approaches to automated spike sorting, but the majority can be broken down into seven main steps [1]:

Filtering - In the first stage, the raw data recorded from a single extracellular electrode is band-pass filtered to remove the frequency components of interference and noise that are out of band. The frequency range of interest is typically 10 - 10,000 Hz [6].

Spike Detection - After filtering, individual APs need to be identified. Nearly all methods detect APs with two main steps: the pre-emphasis of the signal and the use of an amplitude threshold. Pre-emphasis may be performed using methods such as the *non-linear energy operator* (NEO) that attempts to locate rapid changes within the signal [7]. An amplitude threshold based detector is then applied with a threshold that is generally determined automatically using an unsupervised training process [8].

Alignment - Once APs have been identified within the recording, they are extracted by sampling a uniform length window of the original data. At this point each sampled window contains a single AP and, before further processing can take place, the APs must be temporally aligned based on a measure such as the maximum value or the point of maximal slope [9].

Feature Extraction - After alignment, the morphology of the APs is described using a set of *features* such as the maximum amplitude or duration in time [9]. Typically, however, more complex methods employing statistical concepts such as *principal component analysis* (PCA) are used to extract unique features from the captured APs [10].

Dimensionality Reduction - The resulting features may have

This work was supported in part by the Brian Nicholson PhD Scholarship.

B. Metcalfe, C. Clarke and J. Taylor are with the Department of Electronics & Electrical Engineering, University of Bath, Bath, BA2 7AY, UK e-mail: b.w.metcalfe@bath.ac.uk.

N. Donaldson is with the Department of Medical Physics, University College London

Manuscript received XXXX XX, XXXX; revised XXXX XX, XXXX.

many dimensions that make AP classification computationally expensive; one common method to reduce the dimensionality of the features is simply to subsample at regular intervals (there are more advanced statistical methods such as Hartigans dip test [11]). When using PCA as the method of feature extraction, it is common to reduce the number of dimensions considered for clustering by simply truncating to N dimensions, where N is typically 2 or 3 [12].

Clustering - The reduced feature set can now be automatically examined for clusters that correspond to different classes, or types, of AP waveform. Clustering (especially unsupervised) is often the most difficult part of the spike sorting process. The *de-facto* benchmark is *k*-means clustering, but this relies directly on human intervention and so is not suitable for unsupervised learning. One algorithm that is suitable for unsupervised clustering is called *valley seeking* [13], however the algorithm does not operate in real time and has serious drawbacks in terms of complexity. When dealing with clusters that are irregular in shape, improvements may be obtained with *superparamagnetic* clustering [1].

Classification - As new APs are observed in the recorded data, the process is repeated and the clusters are updated and can be used to classify new APs. From this, a visual record called a *spike train* can be produced that shows the firing patterns of each class of AP.

There are two main sources of spurious signals that pose significant challenges in extracellular recordings: deterministic common mode interference from sources such as excited muscles, and random signals such as thermal noise. When noise and interference are present, similar APs originating from different neurons may appear to be the same or, alternatively, APs from the same neuron may appear different. One of the critical constraints of the spike sorting process is the alignment of individual APs before feature extraction is applied. Any temporal misalignment of APs at this time can have a detrimental effect on spike classification. The alignment process is often performed at higher sampling rates in order to minimise sampling jitter and the most common method is to align each AP to the point of *maximum slope* [9]. Other methods consider the point of maximum amplitude or the point (or points) at which the amplitude has been reduced by 3 dB. These methods typically employ only one or two samples from the entire AP and so are highly sensitive to the broadband noise that is typical of biological recordings [6]. Alignment to a metric that is derived from the whole AP rather than from a single point will, in general, be less susceptible to the effects of both background noise and interference and it is an approach based on this principle that is considered in this paper.

Here we describe a new, real-time time approach for spike alignment based on a *centroid* filter that provides an alternative to the traditional spike alignment methods and substantially improves the resilience to noise. An overview of the concept is presented in Section II. We validate the new method by simulation using deterministic models of nerve signals with added noise (both white and red noise models are considered). Further, we show results measured from a hardware implementation. Comparisons are then made between the methods. We demonstrate a power-efficient realisation of the centroid filter that operates in real time on a single Altera Cyclone II *field programmable gate array* (FPGA) in Section III. The use of FPGAs for signal processing tasks is well documented and they are particularly well suited to the rapid analysis and assessment of novel techniques. An FPGA is used here to evaluate the power and resource requirements for the proposed designs that are key metrics for implantable technologies; the results given would apply equally to a full-custom implementation [14].

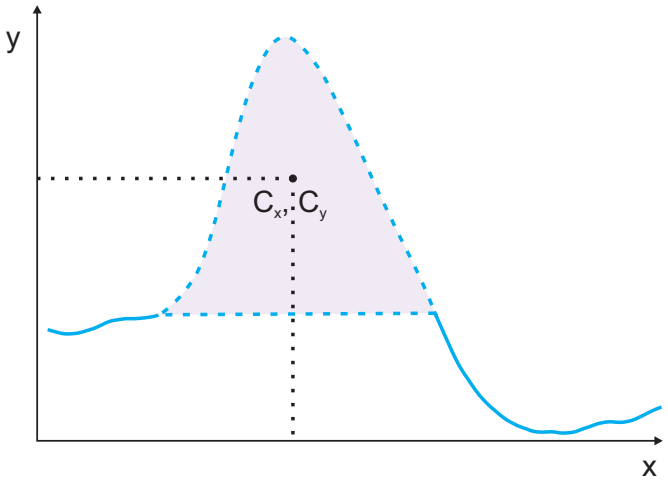


Fig. 1. The concept of transforming a waveform into a polygon. The waveform has been partitioned into a shaded area that may be treated as a plane polygon with centroid located at C_x,C_y.

II. Centroid Filtering Technique

In this section we describe a more robust and effective method that performs temporal alignment of APs based on an average of all of the samples within the AP. If the AP is considered to be a topological shape (i.e. a closed, non-self-intersecting polygon) then the centre of the AP in both time and amplitude can be considered to be the centroid [15], [16]. This concept is illustrated in Fig. 1, where a simple waveform has been partitioned into the shaded area that can be treated as a plane polygon with centroid located at C_x,C_y, which is the centre of mass of the hypothetical polygon.

Because the waveform is considered as a plane figure, the time axis is labelled x and the voltage axis y. It can be regarded as a polygon because the continuous time waveform has been uniformly sampled and may be represented by a series of discrete vectors. Alignment to the centroid has the potential to be substantially more robust to noise and interference than single sample measures, as the centroid is a function of all the samples within the waveform. Consider now the hypothetical time domain AP shown in the solid trace of Fig. 2. The AP can be partitioned into a polygon bound by its width at the points ω_1 and ω_2, and by the intersection with the line $y = 0$, forming a plane polygon with unit density per unit area. In practice, the interception with the line $y = 0$ may be found by half-wave rectification of the waveform. The centroid of

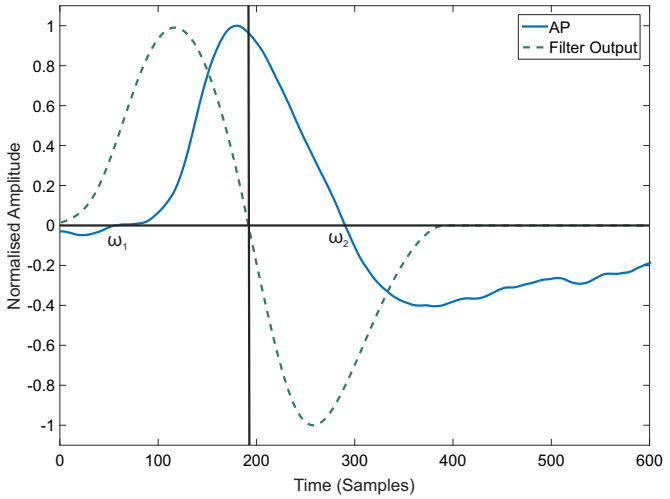


Fig. 2. The application of the centroid filter to a single channel of data containing a realistic AP (solid line). In this case the length of the centroid filter was chosen to be $L = 100$ samples, or approximately the width of the positive phase of the AP. The solid line represents a discrete time version of the AP, and the dashed line the filter output $y[n]$. The vertical marker is set at the negative-going zero crossing of the filter output and is located at the centroid of the AP function along the x axis.

the AP calculated along the x axis can therefore be found by using standard methods, i.e. taking the first moment of area about the axis and dividing by the total area:

$$C_x \equiv \frac{1}{A}\int_{\omega_1}^{\omega_2} xf(x)dx \tag{1}$$

where the centroid calculated along the x axis is C_x, A is the area under the AP, $(\omega_2 - \omega_1)$ is its width and $f(x)$ represents the AP function itself. Consider next the continuous time *convolution integral* of $f(x)$ with another function $h(x)$:

$$y(x) = (h * f)(x) = \int_{-\infty}^{\infty} h(x-\lambda)f(\lambda)d\lambda \tag{2}$$

Note that if this integral is evaluated at the origin ($x = 0$), after a change of variables and a suitable choice of limits, it reduces to:

$$y(0) = \int_{-\infty}^{\infty} h(-\lambda)f(\lambda)d\lambda \tag{3}$$

Inspection of (3) reveals that it is possible to perform the convolution operation using a simple FIR filter. Specifically if the impulse response of the filter is made to be a function such as:

$$h(\lambda) = -k\lambda \tag{4}$$

Then equations (2) and (3) reduce to (1). Considering now the general form of the discrete time output expression $y[n]$ for an FIR filter of length L in the time domain with inputs $x[n]$ and coefficients b_i, the filter can be represented as follows:

$$y[n] = b_0.x[n] + b_1.x[n-1] + ... + b_L.x[n-L] = \sum_{i=0}^{L} b_i.x[n-i] \tag{5}$$

To realise (3) as an FIR filter, we set the coefficients (and thus impulse response) to be:

$$b_i = \frac{-2i}{L} + 1 = mi + 1 \tag{6}$$

i.e. $b_0 = 1$, $b_L = -1$, $m = -2/L$ and where the constant (unity) term is required to make the filter realisable. Such an FIR filter represents a discrete time form of (3), the importance of this result is discussed further in the appendix. A more detailed analysis of the convolution and the filters frequency response is also provided in the appendix.

III. Implementation and Optimisation

It is well known that power consumption is a key metric for any implantable device, therefore the centroid filter that was discussed in Section II is now optimised for implementation in *very large scale integrated* (VLSI) architectures such as FPGAs, where power consumption can be accurately measured. In a direct-form implementation, a length L FIR filter requires L multipliers and L adders, and generally speaking multiplication will dominate the power budget. In this section we describe a method that uses the linear form of (6) to an advantage in order to avoid the use of multipliers as far as possible. We begin by expanding (5) and hence calculate the next output sample:

$$y[n+1] = b_0x[n+1] + ... + b_L.x[n-L+1] = \sum_{i=0}^{L} b_i.x[n+1-i] \tag{7}$$

Recalling that $b_0 = 1$ and $b_L = -1$, then, from (6) the following recurrence relation between the coefficients can be derived:

$$b_i - b_{i-1} = m \tag{8}$$

The expression for the next output value of (7) may now be rearranged as follows:

$$y[n+1] = y[n] + \left\{ m\sum_{i=0}^{L-1} x[n-i] \right\} + x[n+1] - x[n-L] \tag{9}$$

The computation of $y[n+1]$ in this way requires $L+3$ additions and 1 multiplication. Furthermore, if m is a power of 2 the multiplication can be replaced by a left shift. Also, the computation of the summation term in (9) can be accomplished by using a *rolling sum* expressed as follows, where $\omega[n]$ is the rolling sum:

$$\omega[n] = m\sum_{i=0}^{L-1} x[n-i] \tag{10}$$

$\omega[n]$ can be expanded as follows:

$$\omega[n+1] = \omega[n] + m\{x[n+1] - x[n-L-1]\} \tag{11}$$

Substitution of (11) back into (9) results in the following optimised expression for the filtering operation $y[n+1]$:

$$y[n+1] = y[n] + \omega[n] + m\{x[n+1] - x[n-L-1]\} + x[n+1] - x[n-L] \tag{12}$$

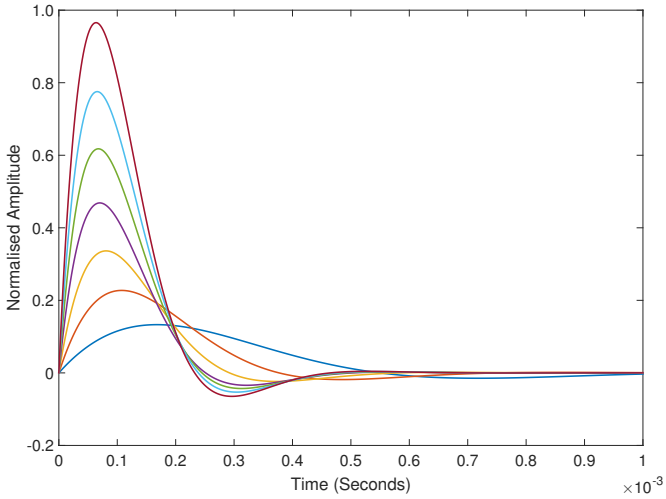


Fig. 3. Seven different SFAPs generated using the damped sinusoid model given in (17). The diameters from the top to bottom are 19, 15, 13, 11, 9, 7 & 5 μm respectively.

The computational effort has been reduced drastically compared to a direct implementation of an FIR filter and an implementation of (12) requires *one* multiplier and *five* adders regardless of the length of the filter (the earlier remark about m being chosen as a power of 2 still applies). The shift register used to store the previous samples is the only component that is dependent on the order of the filter.

IV. Modelling and Simulated Results

A. Single Fibre Action Potential Model

In order to validate the centroid filter as well as to measure the power consumption, simulations were performed based on the deterministic models of the AP and noise functions; the results of which are described in this section. A *single fibre action potential* (SFAP) model was used for generating simulated APs and is described in (13). It is a first-order analytical model of the extracellular AP as recorded from a single neuron and is based on a damped sinusoidal function as proposed in [12] and [17].

$$f(t) = D^2 A sin\left(\frac{t}{\tau_1}\right) e^{\frac{-t}{\tau_2}} \tag{13}$$

where D is the diameter of the axon, t is time and the parameters A, τ_1 and τ_2 determine the amplitude, rising edge rate and duration of each SFAP respectively. The parameters used for a range of different axonal diameters are listed in Table I, having been determined from experimental recordings made from the sciatic nerve of *frog* [17]. The parameters are arranged so that the SFAP amplitude and duration are both a function of the diameter of the axon, for the purposes of simulation a range of different diameters were considered in order to provide a broad selection of different SFAP waveforms. Fig. 3 shows the seven SFAPs generated using this model, where the parameters have been extracted from Table I.

TABLE I
Model Parameters That Define the SFAP Functions Computed Using (13).

Axon Diameter (μm)	**A**	τ_1(ms)	τ_2(ms)
5	2.42	0.175	0.25
7	2.65	0.120	0.15
9	2.73	0.093	0.11
11	2.73	0.080	0.096
13	2.79	0.078	0.092
15	2.80	0.076	0.089
19	2.89	0.072	0.084

B. Noise Models

In all simulations, random noise was added to noiseless SFAPs to produce a signal with a specific, controllable *signal-to-noise ratio* (SNR). As the variance of both the signal and noise were measurable independently, and because both were normalised to zero mean, the following formula was employed to calculate SNR:

$$SNR(\text{db}) = 10log\left(\frac{\sigma^2_{signal}}{\sigma^2_{noise}}\right) \tag{14}$$

The SNR was measured over the entire simulated recording, which lasted 100 ms. Both *additive white Gaussian noise* (AWGN) and *correlated stochastic noise* (CSN) processes were considered since both have relevance to the neural recording problem. For example, AWGN is frequently used in theoretical studies of neural recordings because *myelinated* axons are considered to be electrically isolated and therefore have no synaptic interactions within a nerve [18]. In theory, therefore, background noise in recordings made from hook or suction electrodes of large myelinated nerves could be approximated by a Gaussian noise process. In practice, however, *ephaptic* interactions between axons and crosstalk between amplifiers and recording equipment introduce various levels of correlation [12]. Furthermore any interference from other sources such as muscle activity will appear with some level of correlation. Therefore a correlated noise model was also considered and used to describe the background activity. It was generated using a dynamic *Ornstein-Uhlenbeck* (OU) process described by the difference equation:

$$OU_{t+dt} = OU_t - \frac{OU_t}{\tau}dt + d\Omega_T \tag{15}$$

where OU_t is the value of the noise process at each time step, dt is the simulation time step, τ is the time constant of the process and Ω_t denotes a *Wiener* process [19]. A Wiener process is any process where $\Omega_0 = 0, E[\Omega_t] = 0$ and $\Omega_{t+dt} - \Omega_t \sim N(0, dt)$, where $N(0, dt)$ is a Gaussian with zero mean and variance dt. Fig. 4 illustrates the frequency spectra of the AWGN and OU noise processes for purposes of comparison. The spectra were calculated using the FFT with time domain noise samples of 100 ms in length with a sample rate of 100 kHz and time constant $\tau = 10$ ms. The spectrum of the OU noise process is inherently band limited (*red*) and the noise power of the OU noise process falls below that of the AWGN model at a frequency of approximately 3 kHz.

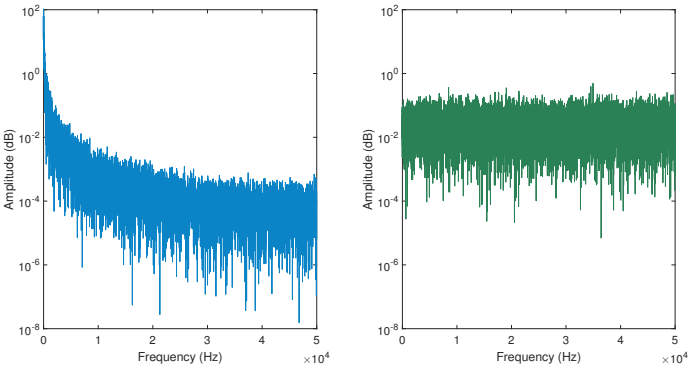


Fig. 4. Frequency spectra over the range DC to 50 kHz for OU (**left inset**) and AWGN (**right inset**) noise processes computed using the FFT over a window of 100 ms. The time constant for the OU process (τ) was 10 ms. The OU noise process follows a *Brownian* or *red* noise profile and is inherently band-limited, by contrast the basic profile of AWGN by definition is flat across the entire range of frequencies.

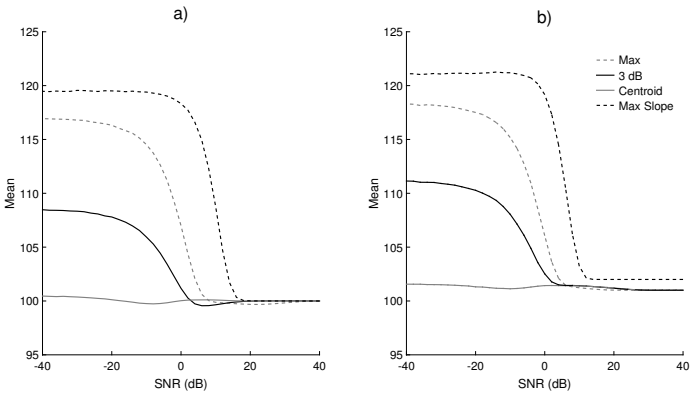


Fig. 5. The mean locations of the four alignment metrics in the presence of AWGN with SNR from 40 dB to -40 dB. In (**a**) no filtering was applied to the noise whereas in (**b**) a low-pass filter with an upper cut-off frequency of 10 kHz was used, which is representative of the first stage of the spike sorting process.

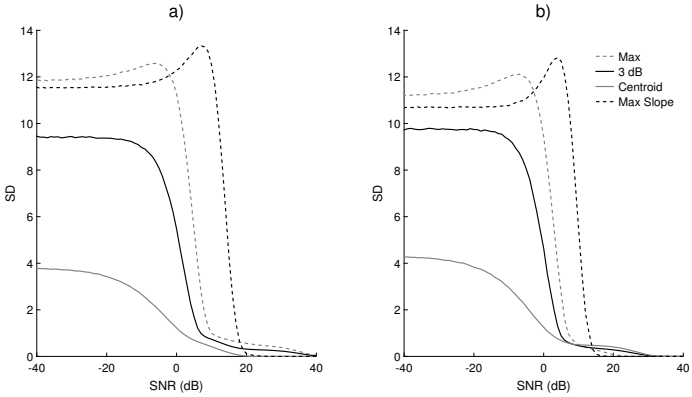


Fig. 6. The standard deviations of the locations of the four alignment metrics in the presence of AWGN with SNR from 40 dB to -40 dB. In (**a**) no filtering was applied to the noise whereas in (**b**) a low-pass filter with an upper cut-off frequency of 10 kHz was used, which is representative of the first stage of the spike sorting process.

The use of a band-limited red noise process as well as the more conventional white noise process is in response to the typical noise profiles encountered in available physiological recordings, which are almost exclusively band limited.

C. Results with AWGN and OU Noise Processes

Simulations to determine the effect of noise on the alignment process were performed using both AWGN and OU noise processes, applied *separately* to simulated SFAPs. In each simulation a single SFAP was generated using a random selection from the parameters given in Table I with a sampling frequency $f_s = 50$ kHz (the bandwidth of each SFAP was approximately 10 kHz, the sampling frequency of 50 kHz was chosen to be approximately twice the Nyquist rate). The SFAP reference location was then calculated from the noise-free model using: the point of maximum slope of the SFAP $P_{maxSlope}$; the global maximum value P_{max}; the mid-point of the -3 dB points P_{3dB}; and the centroid $P_{centroid}$. Once the reference values had been computed, noise was added to the simulated SFAP to produce SNR values from 40 dB to -40 dB in linear steps of 1 dB using either AWGN or OU noise processes. For each value of SNR the experiment was repeated 100,000 times and the SFAP location computed using the four methods. The SFAP locations computed after the addition of noise were then compared to the reference values, and for each SNR the mean and the standard deviation of the detected positions were computed relative to a normalised reference position of 100 samples from the start of the recording for the purposes of comparison and presentation.

The normalized plots shown in Figs. 5 and 6 were obtained by applying these methods using unfiltered AWGN as the noise source. The mean locations and standard deviations of the pulse parameters are plotted for the four alignment methods. For the case of AWGN, in addition to applying the white noise directly, the noise was low-pass filtered, representative of the first part of the spike sorting process. An 8^{th} order Butterworth digital *infinite impulse response* (IIR) low-pass filter with an upper cut-off frequency of 10 kHz was employed. Figs. 5 and 6 are split into two pairs (5a, 5b & 6a, 6b) to accommodate this comparison.

The main feature of these results (i.e. for both mean and standard deviation) is that SFAP location based on maximum slope ($P_{maxSlope}$) appears to be the most susceptible to noise, degrading severely with SNR less than approximately 20 dB. This is followed by the method based on the maximum value of the SFAP (P_{max}), which fails with SNR of approximately 0 dB. The methods based on the mean of the -3 dB points (P_{3dB}) and the centroid ($P_{centroid}$) perform much better, operating well with SNRs of around -10 dB. In these simulations, the centroid method consistently outperformed the others.

In the second case, white (i.e. uncorrelated) noise was replaced by correlated noise generated using the OU model defined in (15) with the time constant $\tau = 10$ ms. The results are shown in Fig. 7a) and 7b) for the mean and standard deviation respectively. The main effect of this change was that metrics ($P_{maxSlope}$) and (P_{max}), which performed worst under AWGN (especially in the case where filtering was *not* employed), performed much better using OU noise. The overall effect was that the spread in performance of all four

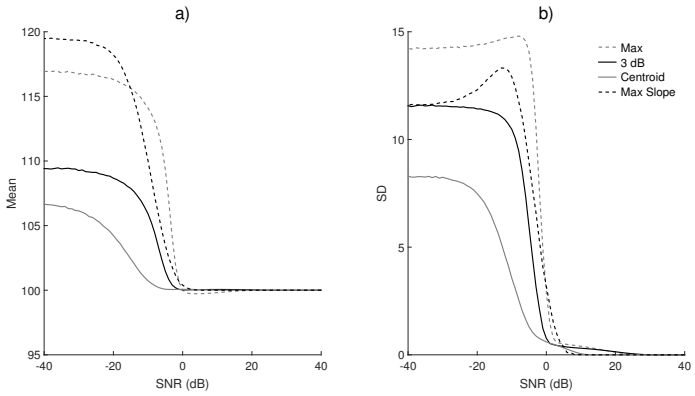


Fig. 7. The mean (**a**) and Standard Deviation (SD) (**b**) for the locations of the four alignment metrics in the presence of OU noise with SNR from 40 dB to -40 dB, no filtering operations have been applied.

TABLE II
COMPARATIVE DEVICE UTILIZATION SUMMARY AND POWER CONSUMPTION FOR A LOW-PASS FILTER AND THE CENTROID FILTER IN AN FPGA, $f_{clk} = 15$ MHz

Method	Parameter	Result
Low-Pass Filter	LC Registers	168
	Logic Cells	2533
	Current Consumption	1.7 mA
	Total Power Consumption (Measured)	2.4 mW
	Total Power Consumption (Estimated)	2.0 mW
Centroid Filter	LC Registers	330
	Logic Cells	442
	Current Consumption	0.18 mA
	Total Power Consumption (Measured)	0.24 mW
	Total Power Consumption (Estimated)	0.25 mW

methods was significantly less than the spread observed with AWGN as the noise source, for both the mean and standard deviation metrics. This suggests that single point temporal measures such as the point of maximal slope or the maximum point are particularly sensitive to high frequency noise components, as are typically present in broadband noise sources. In addition, both these methods rely on determining *turning points* in the time record, emphasising their vulnerability to noise in general, and high frequency components in particular.

D. Power and Area Measurements

An analysis of the power and resource requirements of the proposed centroid filter was performed using dedicated hardware implementations constructed using an FPGA (Altera Cyclone II EP2C35F672C6N FPGA). This device was chosen as a rapid prototyping platform both because of its low cost compared to a custom ASIC and because, at lower sampling rates, an FPGA can consume less power than an equivalent processor-based implementation [20]. Furthermore, the relatively low sampling rates associated with ENG (typically less than 100 kS/s) may allow a reduction in core operating voltages and thus a saving in static power consumption in an FPGA [21]. In order to provide a benchmark, the measurements were duplicated for an 8^{th} order Butterworth IIR filter, this filter was a direct form implementation of the low-pass filter used in section IV. The designs were produced using the SystemVerilog hardware description language and RTL verification was performed using the QuestaSim environment (*QuestaSim 10.2c, Mentor Graphics Inc, Oregon, USA*).

Power measurements were made using a sensitive current meter connected in-line with the appropriate DC power supply ($V_{core} \approx 1.3$V). The synthesis tools (*Quartus 13.0, Altera Corporation, Calif. USA*) reported an initial maximum clock rate for the centroid filter of 25 MHz. The designed clock rate was chosen to be 15 MHz, due to the small nature of the designs under test it was found that power measurements at lower frequencies were unreliable due to measurement errors. The relationship between clock frequency and power consumption is typically linear, and in a full custom implementation the operational clock frequency would likely be at the system sampling rate. In order to make accurate baseline power measurements, clock gating was used to isolate the filter structures, as well as to estimate the power consumed by the I/O and clock propagation circuitry. Further improvements in power measurement were made by implementing multiple instances of each design and dividing the overall power consumption (40 low-pass filters and 350 centroid filters, implemented independently). An overview of the resource and power requirements achieved for a single instance of both methods is provided in Table II. As expected, the low-pass filter consumed more resources and an order of magnitude more power than the centroid filter. The power analysis tools (*Powerplay Power Analyser, Altera Corporation*) were used to predict the power consumption of the designs based on a value dump from simulations and the predicted values were in close agreement with measured data.

E. Verification

In order to verify correct operation of the designs, a set of test vectors was generated and the designs operated at increasing clock speeds on the two devices. Test vectors were generated in MATLAB using the SFAP model (13) before being sampled and transformed into a fixed point format of $Qs0{:}7$, i.e, each sample was represented by a single byte with a sign bit, no decimal bits and 7 fractional bits yielding an approximate range of -1 to 0.992. The RTL codes used to produce designs for the two devices were identical. In addition to the test vectors stored in memory, a verification block was included within the design, this block contained the expected output from the centroid filter and performed simple on-line comparisons on a sample-by-sample basis. The clock control circuitry was used to activate individual modules to obtain accurate power measurements and the verification circuity confirmed correct operation of the centroid filter up to a maximum clock frequency of 24.5 MHz. This frequency is in agreement with the predicted maximum clock frequency of 25 MHz.

V. Discussion

A. General Effects of Noise on Spike Alignment

As outlined in the introduction, neural recording methods have advanced greatly in recent years with the development of multi-electrode methodologies for a wide variety of applications. Of these methods, morphological spike sorting is amongst the most popular. Unfortunately, one of the key stages of such spike sorting methods, temporal alignment, is very susceptible to noise and interference [22]. This is because the methods employed to find the position of individual spikes (APs) in time tend to depend on single point measurements of each AP and are therefore particularly sensitive to the effects of noise processes. This sensitivity was demonstrated in the simulations reported in Section IV of this paper for both uncorrelated and correlated additive noise sources (AWGN and OU noise processes respectively). The basic approach adopted in this paper is therefore to propose a method that does not rely on single point determinations of AP location, but rather employs a measure that, in some sense, is an average of the whole AP - the centroid. As we have demonstrated, this has the effect of reducing the sensitivity of the AP alignment process to additive noise for both correlated and uncorrelated sources.

B. Comparison of Noise Processes

It is generally considered that *myelinated* axons are electrically isolated from each other and have no synaptic interactions within a nerve. It is tempting therefore to assume that background neural activity in a recording is *uncorrelated* and appears without distinguishable spikes. In this way, the interference from nearby axons may be treated as an uncorrelated noise source. Many researchers have adopted this viewpoint and it is common to model both noise and interference as AWGN [18].

However, other workers have found that the spectral characteristics of the neural background are not white, but *red* (Brownian) and can be described as a *correlated* source. In practice, myelination of axons serves to inhibit but not to suppress entirely the *ephaptic* interactions between adjacent neurons, a process that occurs due to the exchange of ions between the cells or as a result of local field potentials. There will also be crosstalk between adjacent amplifiers, and the recording equipment itself will introduce interference with some level of correlation [12].

Therefore, a correlated noise model was also considered in the simulation study, based on the *Ornstein-Uhlenbeck* process [19]. The effect of this change was dramatic on the two methods of spike position determination based on single points in time. These were the point of maximum AP slope and the global maximum point. This was not surprising, because in addition to being single point measures, these methods require the calculation of a turning point in time, suggesting an enhanced sensitivity to high frequency noise components. This view was confirmed by low-pass filtering the signal after the application of AWGN. This is standard practice in spike sorting and the result was comparable to the effect of changing to a correlated source, which is naturally band limited (see Figs. 6 & 4).

C. Benefits of the Centroid Filter

In all cases, and independent of the type of noise or interference applied to the recordings, the centroid proved to be a significantly more reliable metric for aligning the APs than the traditional methods. Intuitively, since the centroid of the AP is a function of the area contained within the AP itself, the addition of *broadband* white noise with zero mean will have little if any effect on the area within the AP. Of course, this simplification does not hold when considering finite time sequences. The results also show that a similar level of performance is obtainable when considering band-limited red noise processes, which are representative of experimental data; indeed the centroid filtering method should be beneficial for any noise process with zero mean. It is possible to apply further processing to the single point metrics, for example several slope values may be obtained by decimating the data stream by different factors and computing an average of the obtained slope values. Such modifications may improve the performance of these methods, especially in the presence of broadband noise. The particular metrics used in this study were chosen as they are considered to be the most commonly used [1].

In its most basic form, as shown in (1), computing the centroid introduces significant over-head in the computation of the integral term, especially in comparison to the single sample measures. However, it has been shown that the computation can be performed with minimal effort using a novel FIR filter structure. A common approach to saving both power and area is the use of rate-adaptive sampling approaches, in this scenario the basic centroid alignment method could still be employed, however the optimised implementation described in Section III would only be valid if the sample rate was constant over the area of the AP (i.e, equation (16) must hold).

It has been shown that the application of a low-pass filter to the raw signal improves the performance of the single sample methods for AP alignment, and in this study an 8^{th} order Butterworth IIR low-pass filter with an upper cut-off frequency of 10 kHz was used both for performance and power comparisons.

D. Implementing the Centroid Filter

When considering implementation costs, the centroid filter, which uses only a single multiplier, will significantly outperform a direct-form IIR filter of almost any order, as the number of multipliers required typically scales with the filter order. More formally, an n^{th} order IIR filter has multiplier complexity $O(n)$, whereas the centroid filter has constant multiplier complexity $O(1)$. Table II details the relative implementation costs for both the centroid filter and the 8^{th} order Butterworth filter used in this study; the Butterworth filter consumes over five times the number of Logic Cells and ten times as much power as the centroid filter. It is important to consider the level of numerical accuracy that is required to perform the filtering operation. In the simulations presented in Section IV, single precision floating point numbers were used throughout, and in the hardware implementation used for comparative area and power measurements, fixed point

number formats were used (floating point multiplication was not required in this example). The effects of finite word lengths in fixed point arithmetic are often problematic, and a highly optimised implementation will often be tightly coupled to the rest of the system characteristics. For example, the overall system sampling rate as well as the word length of the data converters will play a large role on the precision and range required within the filtering operations. The size of the centroid filter will also play an important role, the rolling sum terms of (10), for example, may require the storage of very large numbers for high filter orders. The precision of arithmetic required is minimised by the avoidance (where possible) of multiplication and division, operations that produce fractional bits. In systems with larger filter orders or longer word lengths, it may be necessary to perform the rolling sums using floating point arithmetic. It may be possible to reduce the power consumption further by stopping the filter once the zero crossing event has been detected, however this would remove the ability to detect multiple zero crossings, an event that occurs when over-lapping APs are present in the time window and may be used for automatic rejection of such incidents.

The centroid filter is an efficient and low power method for AP alignment that shows considerable resilience to noise even in SNRs lower than -10 dB. It should be noted that this method has potential applications in a wide range of situations wherever phase information between different pulses is considered, for example in phase-sensitive communication systems.

VI. Conclusions

We have introduced a new method that significantly improves one of the key issues in neural spike sorting: the difficulty of correctly aligning action potentials (APs or spikes) in the time domain in the presence of noise. A novel real-time approach for spike alignment based on a *centroid* filter is introduced that provides an alternative to traditional spike alignment methods and substantially improves resilience to noise and resulting sampling jitter. We have validated the new methods by simulation using deterministic models of nerve signals (i.e. APs with added noise; both correlated and uncorrelated noise models are considered). In addition, we have shown that the new method lends itself particularly well to hardware realisation and a power efficient solution is described that operates in real time on a single Altera Cyclone II *field programmable gate array* (FPGA). The technique has the potential to influence significantly the design of electrophysiological recording systems in the future.

Appendix
Centroid Filter Analysis

The basic operation of the centroid filter can be illustrated using a simple continuous-time approach, considering the convolution of (3) we choose $h(x)$ to be a linear function of x with a negative gradient passing through the origin:

$$h(x) = \frac{-2x}{L} \tag{16}$$

The width of the function $h(x)$ (i.e. the points at which $h = \pm 1$) should be chosen to be greater than the width of $f(x)$, for reasons that will become apparent shortly. Considering a test function $f(x)$ that has unit amplitude and is bounded by ω_1 and ω_1 it can be shown that for values of x where $f(x)$ and $h(x)$ do not overlap, the product of $(h * f)(x)$ is zero. Where overlap occurs, since both functions are discontinuous, a piecewise approach is taken. There are three phases to be considered, depending on the extent of the overlap of the two functions, as $h(x-\lambda)$ approaches $f(x)$ from the left. The first phase occurs as $h(x)$ approaches ω_1 and terminates when the leading edge of $h(x)$ is coincident with ω_2:

$$(h * f)_1(x) = \frac{-2}{L} \int_{\omega_1}^{x+\omega_2 L} (x-\lambda) d\lambda = \frac{(\omega_2 L)^2 - (x-\omega_1)^2}{L} \tag{17}$$

This function is a parabola symmetrical about the vertical axis $x = \omega_1$ and displaced vertically by ${\omega_2}^2 L$. Similarly, the third phase of the process ends when the trailing edge of $h(x)$ makes its last contact with ω_2 and is given by:

$$(h * f)_3(x) = \frac{-2}{L} \int_{x-\omega_1 L}^{\omega_2} (x-\lambda) d\lambda = \frac{(-\omega_1 L)^2 + (-x+\omega_2)^2}{L} \tag{18}$$

Between phases 1 and 3, during phase 2, $h(x)$ completely encloses $f(x)$ since n was chosen to be greater than $|\omega_2 - \omega_1|$. In this phase, the convolution integral evaluates to a linear function:

$$(h * f)_2(x) = \frac{-2}{L} \int_{\omega_1}^{\omega_2} (x-\lambda) d\lambda = \frac{\omega_2 - \omega_1}{L} \left[-2x + (\omega_2 + \omega_1)\right] \tag{19}$$

Equation (19) passes through the point $x = (\omega_2 + \omega_1)/2$, which is the centroid of $f(x)$. So, in the example where $f(x)$ is symmetric about the origin in x ($\omega_1 = -\omega_2$), the convolution integral is simply a straight line passing through the origin with gradient $-2/L$.

In addition, from (17) - (19) it is easy to show that the parabolas of phases 1 and 3 intersect with the linear phase 2 at the points $x_{12} = \omega_2 L - \omega_1$ and $x_{23} = \omega_1 L - \omega_2$. These equations also show the significance of the earlier observation that the function $h(x)$ must be wider than $f(x)$. In the limit where the widths of the functions are equal (i.e. $L = \omega_1 + \omega_2$), the intersection points x_{12} and x_{23} are both zero and the linear phase 2 disappears. The result of this is that a middle zone appears where $y(x) = 0$, the width of this zone is dependent on the location of the intersection points. In this case, it is no longer possible to locate a single zero crossing point to determine the centroid of $f(x)$, however it may be possible to use the first value of $y(x)$ that equals to zero as the alignment point, but this has not been tested. Of course, if the width of $h(x)$ is increased to be much wider than $f(x)$ then it will find the centroid of the combination of $f(x)$ and any noise either side of $f(x)$. Since $h(x)$ can be realised as an FIR filter of

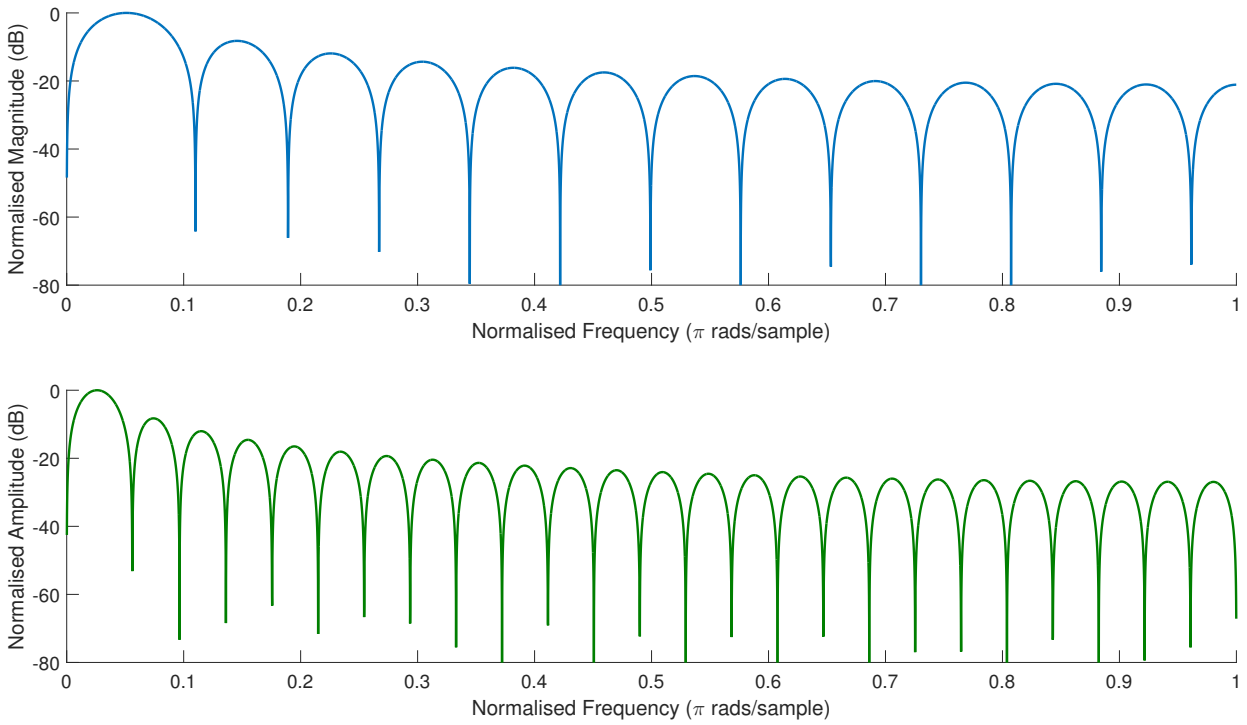


Fig. 8. Frequency response of the centroid filter showing the low-pass nature of equation (23), computed with sampling frequency of $f_s = 50kHz$. **Top inset:** Filter length of 25 samples. **Bottom inset:** Filter length of 50 samples. Increasing the filter length lowers the effective cut-off frequency, but maintains the zero at DC.

length L, given that the relative filter length (thus function width) condition is satisfied, a single filtering operation on $f(x)$, plus zero crossing detection, is sufficient to calculate its centroid. Returning to the example shown in Fig. 2, the result of the filter operation on a single realistic AP (solid line) is shown, the output from the filter crosses the origin at a point coincident with the centroid of the AP (dashed line). Note that the output from the filter goes negative after zero crossing, this corresponds to the third phase of the process as described in (18).

It is interesting to note that the centroid filter can be related to more traditional filter theory by considering it to be the combination of an ideal differentiator $\delta'(x)$ and a low-pass filter $lp(x)$ with a parabolic impulse response, such that:

$$\begin{aligned} y(x) &= \delta'(x) * lp(x) * f(x) \\ &= lp(x) * \delta'(x) * f(x) \\ &= \delta'(x) * lp(x) \end{aligned} \tag{20}$$

Thus the original signal $f(x)$ is differentiated before being smoothed with a simple low-pass filter $lp(x)$. For a simple rectangular test pulse of width $\omega_2 - \omega_1$ this becomes,

$$y(x) = lp(x - \omega_1) - lp(x - \omega_2) \tag{21}$$

which can be read as two parabolas centred on ω_1 and ω_2, and thus comparable to the response of the centroid filter as given in (17) - (19).

It was discussed in Section IV that the centroid filter is particularly effective in the presence of broad band noise, this effectiveness can be explained by considering the frequency response of the filter. The centroid filter has a cut-off frequency that gets lower with increasing filter order, therefore suppressing high-frequency noise components. More precisely, the frequency response of the filter can be derived by considering the product of the impulse response with the rect function, both of length L:

$$h(t) = \frac{-2t}{L}\text{rect}\left(\frac{t}{L}\right) \tag{22}$$

Then by using standard Fourier transform relationships the frequency response H(ω) is found to be:

$$H(\omega) = \frac{2}{j}\text{sinc}'\left(\frac{\omega L}{2}\right) \tag{23}$$

The frequency response was verified numerically for two filters with length $L = 25$ and $L = 50$ samples; these responses are shown in Fig. 8 and illustrate the low-pass nature of the centroid filter as well as the reduction in cut-off frequency with increasing filter length.

REFERENCES

[1] S. Gibson, J. W. Judy, and D. Markovi, "Spike Sorting, the first step in decoding the brain," *IEEE Signal Processing Magazine*, pp. 124–143, Jan 2012.

[2] K. Sameshima and L. A. Baccala, "Trends in multichannel neural ensemble recording instrumentation," in *Methods for Neural Ensemble Recordings*, M. A. L. Nicholelis, Ed. CRC, 1999, ch. 3, pp. 47–60.

[3] K. H. Kim and S. J. Kim, "Neural spike sorting under nearly 0-dB signal-to-noise ratio using nonlinear energy operator and artificial neural-network classifier." *IEEE transactions on bio-medical engineering*, vol. 47, no. 10, pp. 1406–11, Oct. 2000.

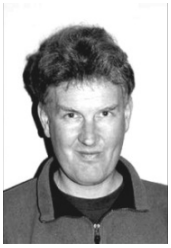
[4] C. M. Stewart, S. D. Newlands, and A. A. Perachio, "Spike detection, characterization, and discrimination using feature analysis software written in LabVIEW," *Computer methods and programs in biomedicine*, vol. 76, no. 3, pp. 239–51, Dec. 2004.

[5] G. Zouridakis and D. Tam, "Identification of reliable spike templates in multi-unit extracellular recordings using fuzzy clustering," *Computer methods and programs in biomedicine*, vol. 61, pp. 91–98, 2000.
[6] N. Donaldson, R. Rieger, M. Schuettler, and J. Taylor, "Noise and selectivity of velocity-selective multi-electrode nerve cuffs." *Medical & biological engineering & computing*, vol. 46, no. 10, pp. 1005–18, Oct 2008.
[7] J. Kaiser, "On a simple algorithm to calculate the energy of a signal," *Acoustics, Speech, and Signal Processing*, vol. 2, no. 10, pp. 381–384, 1990.
[8] S. Mukhopadhyay and G. C. Ray, "A new interpretation of nonlinear energy operator and its efficacy in spike detection." *IEEE transactions on bio-medical engineering*, vol. 45, no. 2, pp. 180–7, Feb. 1998.
[9] M. S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials," *Network (Bristol, England)*, vol. 9, no. 4, pp. 53–78, Nov. 1998.
[10] M. Abeles and M. G. Jr, "Multispike train analysis," *Proceedings of the IEEE*, vol. 65, no. 5, 1977.
[11] J. Hartigan and P. Hartigan, "The dip test of unimodality," *The Annals of Statistics*, 1985.
[12] D. A. Adamos, E. K. Kosmidis, and G. Theophilidis, "Performance evaluation of PCA-based spike sorting algorithms." *Computer methods and programs in biomedicine*, vol. 91, no. 3, pp. 232–44, Sep. 2008.
[13] C. Zhang, X. Zhang, M. Q. Zhang, and Y. Li, "Neighbor number, valley seeking and clustering," *Pattern Recognition Letters*, vol. 28, no. 2, pp. 173–180, Jan. 2007.
[14] S. Narasimhan, X. Wang, and S. Bhunia, "Implantable electronics: emerging design issues and an ultra light-weight security solution." *International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 2010, pp. 6425–8, Jan. 2010.
[15] B. W. Metcalfe, D. J. Chew, C. T. Clarke, N. d. N. Donaldson, and J. T. Taylor, "A new method for spike extraction using velocity selective recording demonstrated with physiological ENG in Rat." *Journal of neuroscience methods*, vol. 251, pp. 47–55, Aug. 2015.
[16] B. W. Metcalfe, D. J. Chew, C. T. Clarke, N. N. Donaldson, and J. T. Taylor, "An Enhancement to Velocity Selective Discrimination of Neural Recordings: Extraction of Neuronal Firing Rates," in *Proceedings of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Aug. 2014, pp. 4111–4114.
[17] N. Dalkiliç and F. Pehlivan, "Comparison of fiber diameter distributions deduced by modeling compound action potentials recorded by extracellular and suction techniques." *The International journal of neuroscience*, vol. 112, no. 8, pp. 913–30, Aug. 2002.
[18] J. Segundo, J. Vibert, K. Pakdaman, M. Stiber, and O. D. Martinez, "Noise and the neurosciences: a long history, a recent revival and some theory," *Origins: Brain and Self Organization*, p. 299, 1994.
[19] L. M. Ricciardi and L. Sacerdote, "The Ornstein-Uhlenbeck process as a model for neuronal activity," *Biological Cybernetics*, vol. 35, no. 1, pp. 1–9, Mar. 1979.
[20] P. Kwan and C. Clarke, "FPGAs for improved energy efficiency in processor based systems," *Advances in Computer Systems Architecture*, pp. 440–449, 2005.
[21] C. Chow, L. Tsui, P. Leong, W. Luk, and S. Wilton, "Dynamic voltage scaling for commercial FPGAs," *Proceedings, IEEE International Conference on Field-Programmable Technology*, pp. 173–180, 2005.
[22] H. K. Jung, J. H. Choi, and T. Kim, "Solving alignment problems in neural spike sorting using frequency domain PCA," *Neurocomputing*, vol. 69, no. 7-9, pp. 975–978, Mar 2006.

Benjamin W. Metcalfe received his PhD and MEng(hons) degrees in Electronic and Electrical Engineering from the University of Bath, UK in 2016 and 2012 respectively. He has worked for Siemens Healthcare designing measurement systems for MRI and has consulted on a number of low-power signal processing projects. He is currently employed as a lecturer at the Centre for Advanced Sensor Technologies (CAST). His research interests are in the field of low-power mixed-signal system design with a focus on the implementation of signal processing for bioelectronic applications. He is a member of the IET, the IEEE and the IMechE and has published multiple papers in the field of bioelectronics.

Christopher T. Clarke received a BEng degree in Engineering Electronics and a PhD degree in Computer Science from the University of Warwick in 1989 and 1994, respectively. From 1994 to 1997, he lectured at Nanyang Technological University in Singapore where he was a co-founder of the Centre for High Performance Embedded Systems (CHiPES). He joined the Microelectronics and Optoelectronics research group in the Department of Electronic and Electrical Engineering at the University of Bath in March 2003. He is a member of the Centre for Advanced Sensor Technologies (CAST). Dr Clarke is a member of the IET and has published in excess of 40 papers in conferences and journals.

Nick Donaldson studied Engineering and Electrical Sciences at Cambridge University. Since 1992, he has been Head of the Implanted Devices Group at University College London. He has been Principal Investigator for many projects related to implanted devices and functional electrical stimulation and is a Professor of Neuroprosthesis Engineering. His research interests now include the development of implanted devices that use natural nerve signals as inputs; stimulators of nerve roots; the use of electrical stimulation for recreational exercise of paralysed legs; and methods to encourage functional neurological recovery after injury.



John Taylor received the BSc and PhD degrees from Imperial College, London University in 1973 and 1984, respectively. In 1984-1985, he worked in the Department of Electrical Engineering, University of Edinburgh, Scotland. He joined the Department of Electronic and Electrical Engineering at University College London in 1985 and subsequently, in 2002, the Department of Electronic and Electrical Engineering at the University of Bath, Bath, UK, where he holds the position of Professor of Microelectronics and Optoelectronics and Director of the University Centre for Advanced Sensor Technologies (CAST). His research interests are in the field of analogue circuit and system design, including low power implantable systems for biomedical and rehabilitation applications. Professor Taylor has published more than 140 technical papers in international journals and conferences.