

An Adaptive Recovery Method in Compressed Sensing of Extracellular Neural Recording

Hicham Semmaoui,
Nan Li,
Saied Hosseini-Khayat,
J.C. Martinez-Trujillo, and
Mahamad Sawan

Polystim Neurotech. Lab, Polytechnique,
Montreal, (QC) Canada

Corresponding author:
Prof. Mohamad Sawan

✉ mohamad.sawan@polymtl.ca

Polytechnique Montreal 2900, Edouard-
Montpetit Montreal (QC), Canada H3T1J4.

Tel: +1 514 340 4711

Citation: Hesse S, Werner C. Transcranial Low-level Laser Therapy may Improve Alertness and Awareness in Traumatic Brain Injured Subjects with Severe Disorders of Consciousness: A Case Series. *J Neurol Neurosci.* 2016, 6:1.

Abstract

A novel adaptive recovery method in the emerging compressed sensing theory is described and applied to extracellular neural recordings in order to reduce data rate in wireless neural recording systems. To strike a balance between high compression ratio and high spike reconstruction quality, a novel method that employs a group-sparsity recovery algorithm, prior information about the input neural signal, learning prior supports of spikes, and a matched wavelet technique is introduced. Our simulation results, using four different sets of real extracellular recordings from four distinct neural sources, show that our proposed method is effective, viable, and outperforms the state-of-the-art compressed sensing-based methods, in particular, when the number of the measurement is two times of the sparsity.

Keywords: Neural recording data reduction; Compressed sensing, Adaptive recovery; Matched wavelet; Spike detection

Received: May 12, 2015; **Accepted:** August 13, 2015; **Published:** August 17, 2015

Introduction

Recording the activity of cortical neurons with microelectrode arrays enables neuroscientists to observe simultaneously the activity of a large number of neurons in the brain, to investigate the function of the nervous system, and to retrieve real-time motor commands that can drive a neuro-prosthesis or a Brain-Machine Interface [1-6].

Currently, an array of microelectrodes can contain as many as tens or even hundreds of microelectrodes, which enables a neural recording device to monitor much larger neuron populations. However, this number of recording units generates a huge volume of data. One of the challenges in the wireless neural recording systems (WNRS) is the correct and efficient transmission of these data out of the body from the implanted device.

The transmission of these large volumes of data requires a high data rate communication link. This presents a significant technical challenge given that extremely low-power dissipation (e.g., ~10 mW) is necessary to avoid heating the surrounding tissue (e.g., <1°C), while small-size requirement (e.g., <1 cm²) prevents the use of an efficient transmitting antenna [2]. For example, for a 100-microelectrode array, with the resolution of 10-bits and the sampling frequency of 20 kHz, data will be generated at an enormous rate of 20 Mb/s.

Given those limitations, several methods have been proposed to reduce data on the implant side prior transmitting them [1-4]. These methods can be mainly divided into two signal processing parts: signal reduction and compression. Signal reduction method can be further divided into spike detection and feature extraction. For example, principle component analysis (PCA) is a widely used method for feature extraction [5]. On the other hand, compressed sensing (CS) technique is a new signal compression method [6]. Comparing both signal processing techniques, signal reduction method keeps some important information and removes the rest of the information of the signal, so original signals cannot be acquired through this method. Signal compression method can recover the original signal after the compression, so this method can largely retain the details of the signal, but, if one wants to use CS technique, the signal should satisfy some criteria (for example, the signal should be sparse). Therefore, both methods have their advantages and drawbacks.

To respect the several constraints imposed on the implant side of the WNRS (e.g., small-size circuit and low-power consumption), our proposed adaptive recovery method employs an emerging compressed sensing (CS) theory. The asymmetric characteristic

of the CS allows the circuit-complexity on the implant side to be moved to the recovery side, which is outside of the body.

Furthermore, in spite of the immaturity of the field, it has been shown that energy efficiency, circuit size, and power consumption of a CS encoder are on par with or better than the state-of-the-art of existing compression methods [7-16].

This paper deals with an adaptive recovery method. This part of a CS system takes place outside of the body. Here, to strike a balance between high compression ratio and high spike reconstruction quality, our proposed method is characterized by a number of unique features: 1) the employment of group-sparsity recovery algorithm, 2) taking advantage of prior information about the input signal, 3) the learning of prior supports of spikes, and 4) a matched wavelet technique.

An overview and comparison of some existing compression methods are given in Aghagolzadeh and Charbiwala [1,6]. However, in this work, in order to further evaluate the performance of our proposed method, we compare it with two recent works in its category (i.e., techniques employing the CS approach).

We recall the basics of the CS in Section II. Section III details our adaptive recovery method. The methodology is presented in Section IV. Then in Section V, our results are reported. An example of a small size and low power cost CS encoder is evinced in Section VI. Finally, we conclude with a discussion of our contribution and perspectives in Section VII.

Compressed Sensing

A. Background of compressed sensing

Compressed sensing is a new approach for signal compression. It has been shown that if a signal has a sparse representation in one basis θ , then it can be recovered from a small number of projections onto a second basis ϕ that is incoherent with the former [17-19]. CS is a non-adaptive data acquisition technique because the basis ϕ does not depend on the measured signal x . Also, to be efficient, CS requires two conditions:

Sparse representation: The sparse property of the signal x is very important and directly influences performance of the CS [17-19]. Given $x = \{x_{(1)}, x_{(2)}, \dots, x_{(N)}\}^T \in \mathbb{R}^N$ (called a frame) and a basis $\psi = \{\psi_1, \psi_2, \dots, \psi_N\}$ for $\mathbb{R}^{N \times N}$. Consider the transform $x = \psi\theta$, where θ represents the $N \times 1$ transform coefficients vector, and ψ is an orthonormal sparsifying basis. We can say that x is K -sparse (approximately K -sparse) if only K ($\ll N$) coefficients of θ are nonzero (significant).

In this case, the CS theory maintains that just a few number of measurements $M = O(K)$ will both preserve all of the signal information and enable robust signal recovery provided that a suitable signal recovery algorithm is used [17-19]. CS aims to provide a reconstruction of $x \in \mathbb{R}^N$ based on measurement $y \in \mathbb{R}^M$, $M \ll N$ obtained by using a sensing matrix $\phi \in \mathbb{R}^{M \times N}$. One can consider the following model:

$$y = \phi x \quad (1)$$

Incoherence: The sensing matrix ϕ should be highly incoherent with the sparsifying matrix ψ , i.e., the rows of ϕ do not have any sparse representation in the matrix ψ . The incoherence between

the two matrices is mathematically quantified by the restricted isometry property or by the mutual incoherence property [17-19]. A popular family of sensing matrices is the set of matrices with random (Gaussian or Bernoulli) entries [17,18]. This family of sensing matrices is well known as it is universally incoherent with all other sparsifying bases [17,18].

Another essential component of CS is the recovery algorithm. Commonly used strategies to recover the frame x are often based on convex relaxation, nonconvex local optimization, or greedy search strategies [20-26].

Convex relaxation is used in algorithms such as basis pursuit and basis pursuit denoising [20], or the lasso [21]. Nonconvex local optimization procedures include the focal underdetermined system solver [22], or sparse Bayesian learning [23]. Finally, the most important recovery algorithms among greedy search strategies are matching pursuit (MP) [24] and orthogonal matching pursuit (OMP) [25]. MP is an algorithm that is often used for practical applications and there are now very efficient implementations of it. OMP has superior performance; however, its implementation is relatively more demanding both in terms of computation time and memory requirement. Recently, the CoSaMP algorithm is introduced [26]. It is based on MP, and incorporates ideas from the CS to accelerate the execution time and to guaranty better recovery.

In this work, we are interested in greedy methods because of their low complexity and simple implementation, which allows their use in real-time applications.

B. Compressed sensing for extracellular neural signal

Recordings of neural signals using a single extracellular electrode or an array of microelectrodes are a mixture of spikes fired by neurons located near the electrode (i.e., single-unit and multi-unit activities) and the background noise (**Figure 1a**).

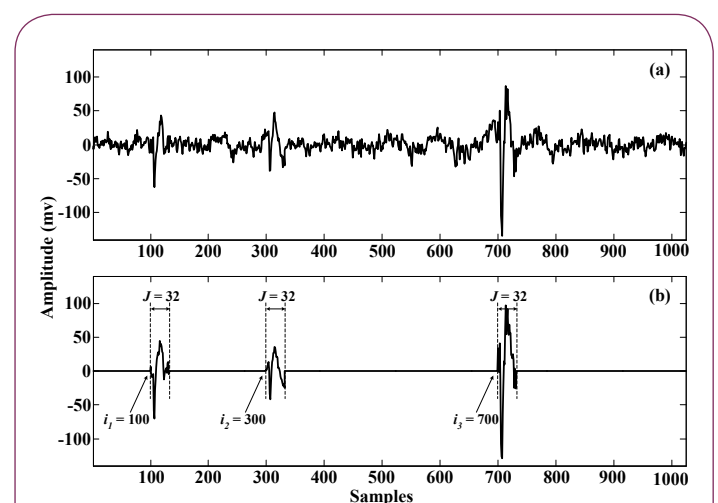


Figure 1 Frame of 1024 samples of recordings from the visual cortex of a rat. (a) Spikes and noise. (b) Result obtained by spike detector, with spike duration J and times of occurrence of detected spikes (i.e., i_1 , i_2 , and i_3).

The background noise is mainly composed of spikes from neurons far from the recording site; hence, spikes and background noise are relatively highly correlated [1,27]. Additionally, the signal to noise ratio (SNR) of extracellular neural signal is usually low. Consequently, this high correlation combined with low SNR makes the neural signal to be not enough compressible in a basis \emptyset . This does not allow a significant reduction of the number of measurements M .

The notion of sparsity/compressibility is critical to the emerging CS theory. It is one of the main measures of signal complexity and plays roughly the same role in CS as that played by bandwidth in the classical Shannon-Nyquist theory. To impose the sparsity requirement in the CS theory, one can be inspired by the idea of anti-aliasing filters. One solution consists in enhancing the SNR and reducing the correlation between the signal x and the background noise before applying CS. There exists several techniques to reach this purpose, such as using a whitening filter [27]. However, in line with the need for simplicity and energy-efficiency [28,29], we employ a simple spike detection technique to increase the sparsity of the frame x . **Figure 1b** shows an example of result obtained by the spike detection technique. The integers i_1 , i_2 , and i_3 are the times of occurrence of the detected spikes.

As mentioned above, CS relies on the principle of sparsity to accomplish sensing and recovery efficiently. Moreover, it has been shown that if the K coefficients (i.e., x is K -sparse) of a frame are clustered into groups (e.g., **Figure 1b**), the required number of measurements M can be reduced [30,33]. To show this fact, consider the frame in **Figure 1b**. By assuming that the frame x is compressible (i.e., nearly K -sparse), we can consider two models to determine its $N \times 1$ wavelet transform coefficients θ . Hence, **Figure 2a** depicts $\hat{\theta}_1$, the 24 significant wavelet coefficients of θ of the frame shown in **Figure 1b**. **Figure 2b** depicts $\hat{\theta}_2$, the 24 (3×8) significant coefficients of θ of the same frame. In the latter, each detected spike is transformed separately; for each one we kept just 8 significant wavelet coefficients, and we sorted each group in decreasing order of magnitudes (**Figure 2b**). We employ the Symlet-2 wavelet transform for the both models, which has been shown to be suitable for neural spikes compression [1]. **Figure 2c** depicts an example performance of the signal recovery of the frame x in **Figure 1b**.

From this example, one can see the potential performance gain (i.e., $M=48$ rather than 80) when we exploit the fact that the K significant coefficients for neural signal are clustered into n groups of length S (where $K=nS$), with n being the number of detected spikes in the frame, and each spike is approximated by S significant wavelet coefficients only.

In CS theory, such model is called group-sparsity signal and it was demonstrated that in order to recover such signals, only a reduced number of measurements M is needed while preserving computational efficiency and robustness to measurement error [30-35].

Spike Reconstruction Using our Adaptive Recovery Method

To recover the 'right' frame x with just a few number of

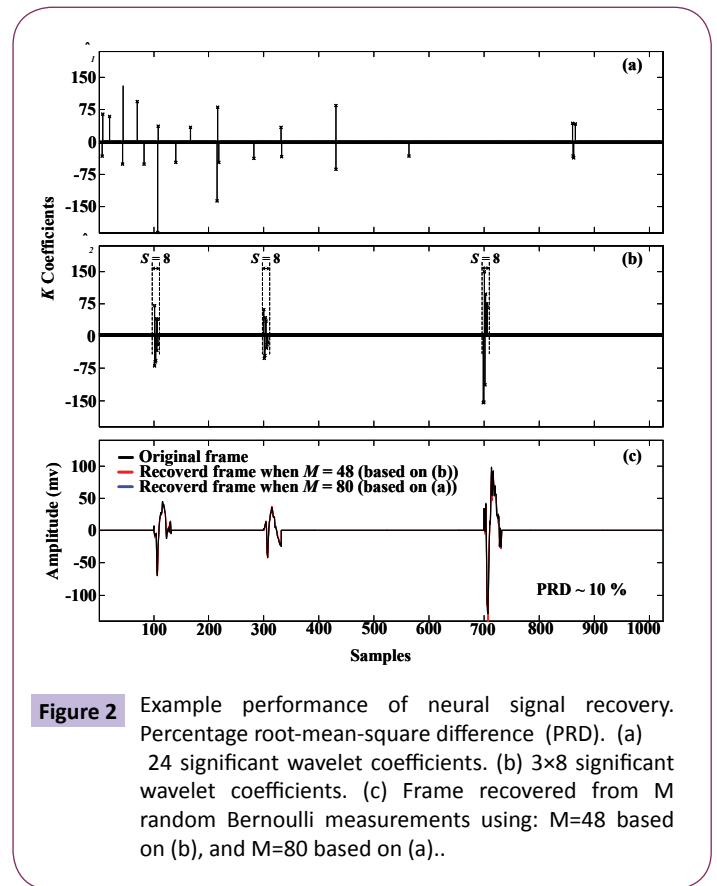


Figure 2 Example performance of neural signal recovery. Percentage root-mean-square difference (PRD). (a) 24 significant wavelet coefficients. (b) 3×8 significant wavelet coefficients. (c) Frame recovered from M random Bernoulli measurements using: $M=48$ based on (b), and $M=80$ based on (a)..

measurements M (i.e., ideally as close to K as possible), one must exploit prior knowledge about the frame in addition to its sparsity or compressibility. Thus, our proposed spike recovery method is adaptive in the sense that it exploits prior knowledge about the frame. However, to allow the use of a 'generic' CS encoder circuit (see Section VI), we keep the CS sensing part at the transmitter non-adaptive and not dependent on x .

In **Figure 3**, the dashed-line part of the block diagram depicts our adaptive recovery method introduced in this work. As shown in **Figure 3**, our adaptive recovery method combines: 1) the dynamic group-sparsity (DGS) recovery algorithm, 2) prior information about the frame, 3) learning of prior supports of spikes, and 4) a matched wavelet technique. We will explain each of the constituent parts of this system in the next sections.

A. The dynamic group-sparsity (DGS) algorithm

In recent years, signals with group-sparsity, whose nonzero coefficients occur in clusters have received much attention. The group-sparsity recovery algorithms (GSA) attempt to utilize these group clustering prior for better sparse recovery [30-35]. Some attempts have been made, most of which assume that a signal of length N (a frame) can be viewed as a concatenation of m clusters (groups) of length J , where $N=mJ$, with m an integer [30]. Another work assumes that the clusters are sufficiently separated in time. In particular, any two consecutive clusters in the frame are separated by at least P locations, where $P > J$ [33]. However, in neural signals, the occurrence of spikes is randomly distributed in time and we have no guarantee that the occurrence of spikes

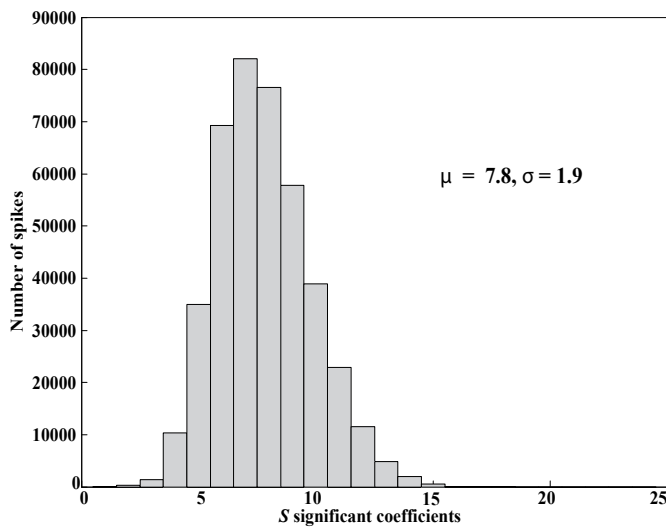


Figure 4 Histogram of the number of S significant coefficients in the DWT domain with Symlet wavelet of order 2 and C=0.95 (3) of over 400,000 spikes, with J=32, extracted from four different sets of extracellular recordings.

Δ_{s_i} to create a $J \times J$ wavelet matrix $\psi_{s_i}(2)$ as below:

where the first S columns of ψ_{s_i} are selected from ψ_s based on Δ_{s_i} ; the remaining $(J - S)$ columns are simply set to 0.

However, the support Δ_{s_i} , corresponding to each detected spike s_i in a frame, is not known in advance. Instead, we know that a

spike s_i belongs to one of $\binom{J}{S}$ subspaces of R^J . Note for example that if the length of a detected spike is $J=32$ and $S=8$, the number of possible subspaces for a detected spike is 10518300, which is an enormous number to search in real-time applications. Fortunately, the supports of spikes detected from the same neuron are very similar. The latter fact is the basis for spike sorting. Thus, the number of possible subspaces is much smaller

than $\binom{J}{S}$ if a prior learning of spike supports is performed. Based on this concept, we propose a simple and iterative approach using the mean squared error (MSE) to approximate the best support of each detected spike in a frame during the recovery step. One conclusion can be acquired from **Figure 4** that S (the number of significant coefficients per spikes in a frame (e.g., $S=8$)).

Our approach is divided into two phases. During the first phase, one learns the support sets Δ_{s_j} (this phase occurs during the learning mode of our adaptive method). During the second phase, the support information is used to recover spikes (this phase occurs during the compression mode).

The approach can be summarized as:

Phase 1: Learning the set of supports

- i. Assume D spikes ($s_i, i=1, 2, \dots, D$) are detected during the learning mode, with each spike having a length J. Each of the D spikes is projected into the $J \times J$ wavelet matrix ψ_s .
- ii. Choose a fixed value of S (e.g., $S=8$). For each s_i , it needs to sort the J coefficients in decreasing order of magnitudes and

select the S largest coefficients. Also, it needs store their locations as the support Δ_{s_i} of spike s_i . This support will be used to create a $J \times J$ wavelet matrix ψ_{s_i} associated with spike s_i as shown in (4).

- iii. Remove the repeated identical supports. Hence, the size of the set of different supports becomes $D' \ll D$.

Phase 2: During the recovery phase

For each $\Delta_{s_j}, j=1, 2, \dots, D'$, in the set of supports, do:

- i. Build a $J \times J$ wavelet matrix ψ_{s_j} as in (4).
- ii. Build $\psi(2)$ by inserting the Δ_{s_j} at all positions i_1, i_2, \dots, i_n (corresponding to the times of occurrence of the n detected spikes in frame x) (**Figure 3**)

Recover the vector $\hat{\theta} \in R^{1 \times N}$ based on the DGS algorithm (as shown in **Figure 3**).

- iii. Extract from the vector $\hat{\theta}$, the n groups corresponding to the n detected spikes. The length of each group u_i is S. The groups are as follows:

$$u_1 = \hat{\theta}(i_1), \hat{\theta}(i_1 + 1), \dots, \hat{\theta}(i_1 + S - 1),$$

$$u_n = \hat{\theta}(i_n), \hat{\theta}(i_n + 1), \dots, \hat{\theta}(i_n + S - 1)$$

$$u_n = \hat{\theta}(i_n), \hat{\theta}(i_n + 1), \dots, \hat{\theta}(i_n + S - 1).$$

For each group u_i , we can sort its elements in decreasing order of magnitudes $u_{i(1)} \geq u_{i(2)} \geq \dots \geq u_{i(S)}$ and denote the result as $v_i = u_{i(1)}, u_{i(2)}, \dots, u_{i(S)}$.

- iv. Compute the MSE_{j_i} between each u_i and v_i .

At the end of this procedure we will build a matrix of MSE_{j_i} of size $D' \times n$. Each column is composed of MSE's of the same detected spike s_i in frame x. The minimum MSE in column will correspond to the best basis ψ_{s_i} of s_i . For each of the n detected spikes in frame x, we use its best basis ψ_{s_i} (allowing the minimum MSE in nth column) to build the sparsifying $N \times N$ matrix $\psi(2)$ which is used to recover the vector $\hat{\theta}$ and for the final recovery of the frame \hat{X} (**Figure 3**).

D. Matched Wavelet

In the procedure described in the previous section, we have assumed that a mother wavelet (i.e., to build the $J \times J$ wavelet matrix ψ_s) is known. In this section, we describe how to determine the optimal mother wavelet that is used to build the matrix ψ_s which increases the sparsity of detected spikes. A topic of research interest is to find a wavelet that can provide the best sparse representation for the signal of interest. Several algorithms have been recently proposed to design a wavelet matched to a signal [37,38]. However, most of these methods are computationally intensive. In line with the need for simplicity, to allow a real-time application, a signal-dependent selection of the mother wavelet based on the criterion C (3) from [39] is adopted in this work. The mother wavelet can be parameterized through a scaling filter h of length F [39]. In particular, if $F=4$, just one single parameter α is needed for the design of h (5) [39]. This allows faster computation than optimization of multiple parameters. The coefficients of the scaling filter h are as in (5) [39].

To choose the optimal parameter α : 1) we record D detected

spikes s_i , $i=1, 2, \dots, D$; 2) we vary α between $-\pi$ and π with a step of $\pi/10$ to design the filter h ; and 3) for each h , based on the criterion C (3), we determine the number of the S coefficients that best approximate each spike s_i within the D recorded spikes.

$$\begin{aligned} h[0] &= (1 - \cos(\alpha) + \sin(\alpha)) / (2\sqrt{2}) \\ h[1] &= (1 + \cos(\alpha) + \sin(\alpha)) / (2\sqrt{2}) \\ h[2] &= (1 + \cos(\alpha) - \sin(\alpha)) / (2\sqrt{2}) \\ h[3] &= (1 - \cos(\alpha) - \sin(\alpha)) / (2\sqrt{2}) \end{aligned} \quad (5)$$

The value of α resulting in the lowest statistical moments (first criteria, the mean μ ; and second criteria, the standard deviation σ) is considered as the optimal one (Figure 4). This is because if we minimize those two moments, the sparsity is maximized. Note that, the decomposition level is set to 4 (because, typical action potentials have frequency range between 1 kHz to 5 kHz) [40].

Once α is selected, we apply the learning of prior supports of spikes procedure described in the previous section on the same D recorded spikes.

Materials and Methods

All the algorithms and data analysis procedures were implemented in MATLAB (Mathworks, Natick, MA). In order to evaluate our proposed method, simulations using synthetic neural signals constructed from real neural recordings were conducted. Four different sets of extracellular recordings from four distinct sources were used for this purpose.

The first data set is recorded from an adult male monkey (Macaca Mulatta) at Cognitive Neurophysiology Laboratory of McGill University. The data contain thirty-two extracellular channels recorded using the Utah 10x10 microelectrode array implemented in the prefrontal cortex. The data consist of three different recordings over three trials. The duration of each trial was 300 seconds. Data were first filtered using a 3rd-order band-pass Butterworth analog filter with cut-off frequencies of 0.3 Hz and 7 kHz. The filtered data were then amplified with a gain of 80 db, sampled at 30 kHz, digitized (10 bits per sample), and finally stored on a computer.

The second set of data is recorded from the visual cortex of a rat at the Center for Studies in Behavioral Neurobiology of Concordia University using a stainless-steel-tipped microelectrode with a shank diameter of 75 μm . The data were filtered using a 4th-order band-pass Butterworth analog filter with cut-off frequencies of 0.15 and 10 kHz. The filtered data were then amplified with a gain of 100 db, sampled at 32 kHz, digitized (10 bits), and finally stored. The recording has a duration of 60 seconds.

The third set is offered by [41]. It consists of a simulated extracellular signal from recording in a human medial temporal lobe using intracranial electrode. The signal is 10 seconds long, and it was amplified, sampled at 32 kHz and filtered between 0.3 and 3 kHz, digitized (12 bits), and stored.

Finally, the fourth set of data is extracted from recordings in a macaque parietal cortex. The data contain four extracellular channels which were recorded using a single Tetrode. The

data are 38 seconds in duration. They were amplified, band-pass filtered (0.3 and 10 kHz), sampled at 20 kHz, digitized (12 bits), and stored. The extracellular recording presented here is retrieved from [42].

To obtain similar recording conditions, all the data were first re-filtered with a second-order non-causal Butterworth band-pass digital filter with cut-off frequencies of 0.3 and 7 kHz (the non-causal filter relatively preserves the shape of waveforms spikes), re-quantized with 10 bits per sample, and finally, re-sampled at 20 kHz. Since, the neural spike length is assumed ~ 1.6 ms [40], then we allocate 32 samples per spike.

In an attempt to evaluate our proposed method, first we extracted free noise segments from the available data to construct background noise libraries. To build large data sets of the background noise, neural noise were modeled by an autoregressive (AR) model [43]. The order of the AR model was determined using Akaike criteria [43], and the AR model coefficients were determined by solving the Yule-Walker equations [43]. We used the fifth-order AR model for the first and third data sets, and the eighth-order model for the second and fourth ones. Second, for each set of data, spikes were detected by setting a threshold manually (above the perceived noise level), isolated (32 samples per spike), aligned, and stored to build spike libraries.

To construct the synthetic neural signals, the extracted spikes from each set of data were randomly inserted into the corresponding noise libraries. Spike timings were generated between 10 and 200 spikes per second (which correspond to $n=0$ to 10 spikes per a frame x of length $N=1024$ as depicted in Figure 5). The SNR of the synthetic neural signals, defined in (6) [1], was fixed randomly between 3 and 6.

$$SN = \frac{\text{Average spike peak to peak amplitude}}{RMS_{\text{Noise}}} \quad (6)$$

Finally, the spike detection stage in Figure 3 is performed by the absolute value (Abs) technique [28]. We selected this method for its relatively good performance and simplicity [29].

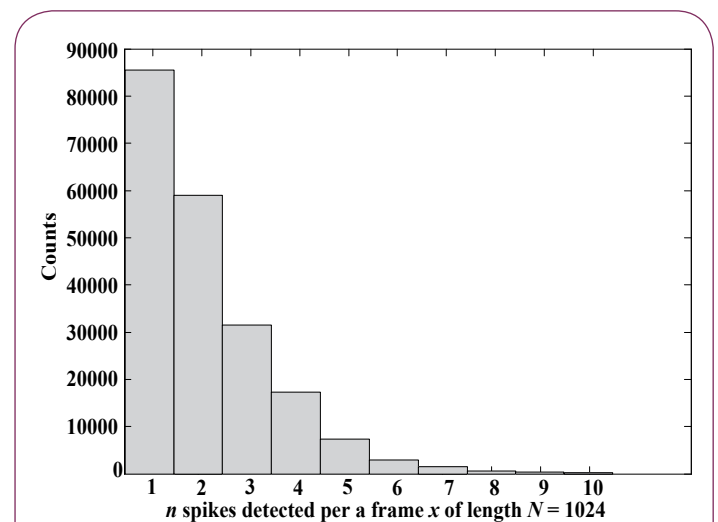


Figure 5 Histogram of the number of the detected spikes per a frame x of length $N=1024$ of over 400,000 spikes from the four sets of neural recordings.

Simulation results

In order to evaluate our proposed method depicted in **Figure 3**, several sets of simulations have been performed. Simulations were separately run for all data sets presented in Section IV. For each one, we calculated the average of the results of a Monte Carlo simulation over 500 trials. Each trial was conducted by extracting from a synthetic neural signal, presented in Section IV, a frame x of length $N=1024$, performing spike detection (Abs technique), computing M linear random Bernoulli measurements (where each entry of the matrix ϕ is \hat{x}_i), recovering the frame \hat{x} , and recording the magnitude of the recovery error for different values of the ratio M/K (from 0.5 to 8 with a step of 0.5). Also, we define B_x and B_y as the bits needed to represent the dynamic range of each sample in x and y respectively (1). We set $B_x=10$ and $B_y=16$, the choice of $B_y=16$ is based on our simulation results. Thus, the effective compression ratio (CR) is $(N \cdot B_x)/(M \cdot B_y)$.

In this work, the recovery error is estimated by the percentage root-mean-square difference (PRD) and the PRD is shown in (7),

$$PRD = \sqrt{\|x - \hat{x}\|_2^2 / \|x\|_2^2} \times 100\% \quad (7)$$

where x is the original signal and \hat{x} is the reconstructed signal.

Note that the sparsity number $K=nS$, where n is the number of detected spikes in a frame, and S is the length of a group (i.e., the number of the S significant coefficients per spike sorted in decreasing order of magnitudes uses our iterative method based on the MSE). In the rest of this paper, frame is referred to the frame x of length $N=1024$ after the spike detection operation.

Figures 6-8 display the average of all data sets.

To increase the sparsity of spikes in the basis ψ_s (4), and to achieve recovery of these spikes with a high accuracy, we used the matched wavelet method and the learning of prior supports of spikes as explained in Section III. Our proposed method operates in two modes. The first mode is called the learning mode, in which we allocate $S=32$ coefficients per detected spike in a frame x (**Figure 6**). This mode is used to build the set of supports and to select the parameter α with $C=0.95$ and $D=1000$. The second mode, called the compression mode, uses the iterative method based on the MSE explained in Section III-C. **Figure 6** shows the

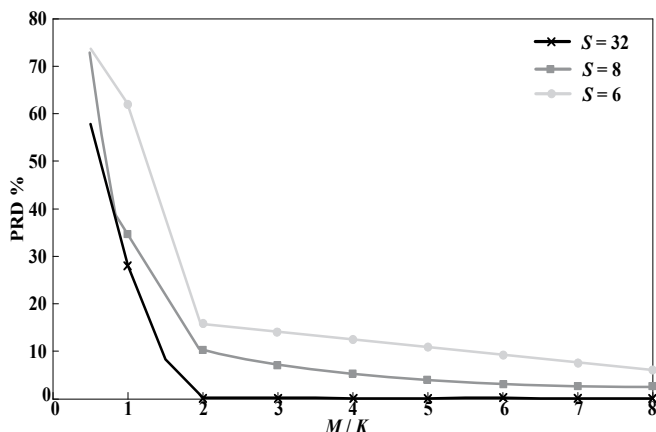


Figure 6 Average PRD versus M/K for $S=6, 8$, and 32 .

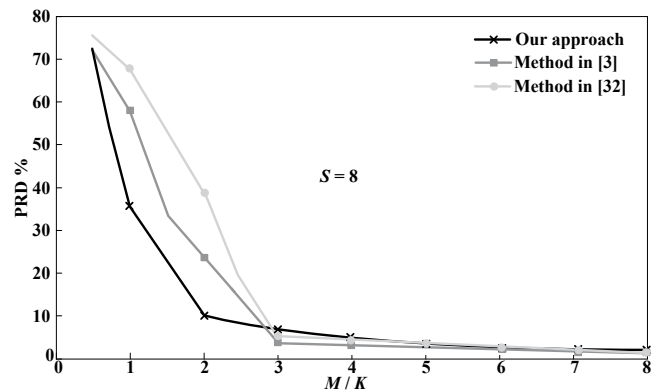


Figure 7 Average PRD, when $S=8$, versus M/K for our method and methods in [3] and [32].

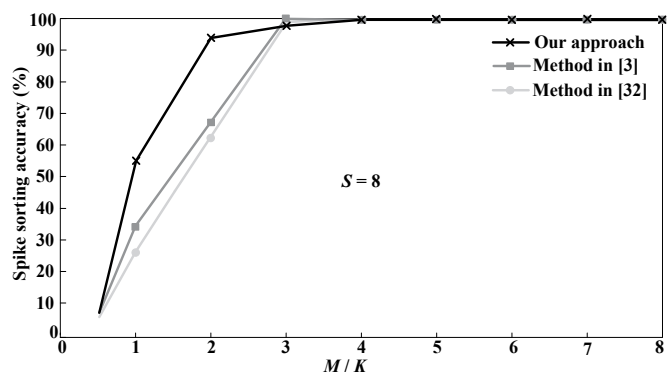


Figure 8 Average spike sorting accuracy over 500 trials, when $S=8$, versus M/K for our method and methods in [3] and [32].

recovery errors when the group lengths are $S=6, 8$, and 32 .

Figure 6 shows that if a frame is sensed with $M=2K$ measurements, the recovery of spikes can be done without any distortion during the learning mode. Also, the distortion observed during the compression mode is relatively small, i.e., the PRD is $\sim 10\%$ and 15% when $S=8$ and 6 , respectively.

Based on **Figure 7**, it can be seen that when $M=2K$, our proposed method achieves a better recovery performance than the methods proposed in [3,32]. The opposite is true when $M \geq 3K$. Remember that ideally one should get $M=K$.

To further evaluate our proposed method, its performance in terms of spike sorting accuracy was estimated and compared against the two methods in [3,32] for different values of the ratio M/K (from 0.5 to 8 with a step of 0.5).

Towards this purpose, we applied the spike sorting method proposed in [44] to the recovered spikes resulting from the three methods. The method in [44] combines DWT-processing with super-paramagnetic clustering. The spike sorting accuracy of the three methods was computed by comparing the classification of spikes resulting from each of the methods against the 'right' classification (i.e., spike sorting of the original spike). **Figure 8** depicts the average results of a Monte Carlo simulation over 500 trials of 100,000 spikes from the four different sets of extracellular recordings.

Figure 8 shows that when $M=2K$, the spike sorting accuracy of

our method is better than the accuracy of the methods in [3,32]. The performance of the three methods are practically the same when $M \geq 3K$.

Example of CS encoder

An idea of the architecture of a CS encoder using a matrix generator ϕ with coefficients c_{ij} is depicted in **Figure 9**. Generation of the matrix ϕ requires a pseudo-random bit sequence (PRBS) generator [8]. This architecture allows the sensing of data stream without using any storage for the incoming data or for the matrix coefficients [8].

At this stage, one important question to be addressed is: How can one determine the minimum number (M) of branches in this encoder (**Figure 9**) leading to a minimum size circuit and better energy-efficiency?

Let M_{op} denote the minimum M . Based on **Figures 6-8**, we can observe that when $M/K=2$, our proposed method offers a good trade-off between recovery performance (quantified in this work by both PRD and spike sorting accuracy (**Figures 7 and 8**)) and compression ratio CR. Recall that $K=nS$, where n is the number of detected spikes in a frame. Hence, M_{op} can be estimated as (8):

$$M_{op} = 2n_{op}S_{op} \tag{8}$$

where n_{op} and S_{op} are the minimum of n and S respectively.

The choice of n_{op} is more delicate; it depends mostly on the

system under investigation and the user's choice. From **Figure 5**, it can be seen that for the data sets presented in this work, the probability of occurrence of $n>8$ spikes per frame is close to zero. Moreover, from **Figures 6 and 8**, it is seen that when $S=8$, the observed distortion and the spike sorting error during the compression mode are relatively small. Therefore, $s=8$ can be a good choice.

However, the choice of $n_{op}=4$ favors small size circuit, better energy-efficiency, and less power consumption (**Table 1**). So, the number of branches M in the sensing circuit when $n_{op}=8$ (4) can be set at 128 (64). **Table 1** depicts a comparison between the power cost and size of the CS-encoder in both situations, i.e., $n_{op}=4$ and $n_{op}=8$.

Note that if one chooses $M_{op}=128$ (64), this does not mean that the encoder will not process situations when more than $n_{op}=8$ (4) spikes were detected in a frame x . It only means that PRD would degrade (**Figure 10**). Contrariwise, when $n<n_{op}$, a higher signal compression ratio can be achieved. For example, if $n=3$ and $S=8$, only $M=2 \times 3 \times 8=48$ measurements will suffice in order to obtain $CR=(1024 \times 10)/((48 \times 16)+3) \sim 14$ and $PRD \sim 10\%$.

From **Figure 10**, one can see that the variability of FR of spikes was taken into account to optimize the recovered spikes quality, while keeping CR to an acceptable level even if FR increases, e.g., $CR \sim 5$ ($CR \sim 10$) when $n>8$ (>4).

Finally, neural signals has nonstationary properties [45]. Therefore, the set of supports and the parameter α need to be refreshed. One method to refresh them is to trigger the learning mode periodically, and the frequency of this operation depends mostly on the system under investigation and user choice.

Discussion and Conclusion

Results presented in this paper illustrate that our proposed method allows achievement of a compromise between high compression ratio and good recovery quality, while keeping CR at an acceptable level even when FR increases (**Figure 10**).

Also, our results demonstrate that the proposed method is more efficient, in regard to the trade-off between CR and spikes recovery quality, than recent CS-based methods [3,32]. We gained this advantage by using the DGS recovery algorithm combined with the matched wavelet technique and learning of prior support of spikes. Certainly, one can use other non-GSA recovery algorithms such as CoSamp or OMP. However, to get better results, we suggest using DGS, as described in this paper. In fact, we replaced DSG with CoSamp and OMP (results not show here) and found out that CoSamp performs better than OMP but not as well as DGS.

Note that, there are special situations that our method needs to deal with:

Table 1 Power cost and size circuit for the CS-block (**Figure 3**).

	ABS-detector [29]		CS-encoder [8]		CS-block	
	Size (mm ²)	Power (μW)	Size (mm ²)	Power (μW)	Size (mm ²)	Power (μW)
M=128	0.0003	0.01	0.0036	~ 4	0.0039	~ 4
M=64	0.0003	0.01	0.0036	~ 2	0.0012	~ 2

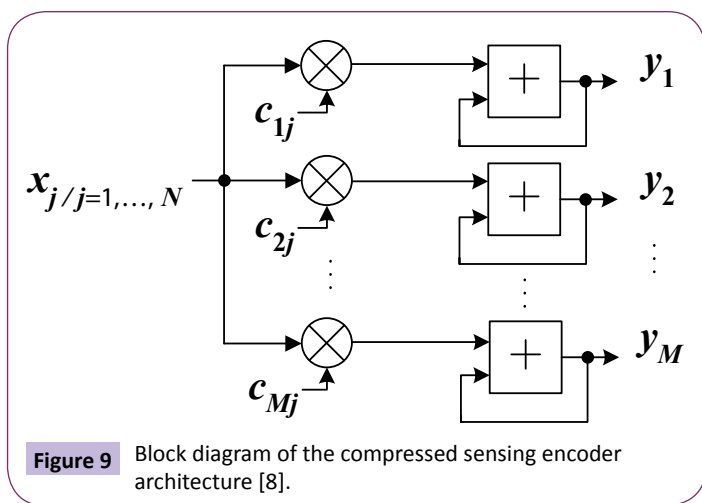


Figure 9 Block diagram of the compressed sensing encoder architecture [8].

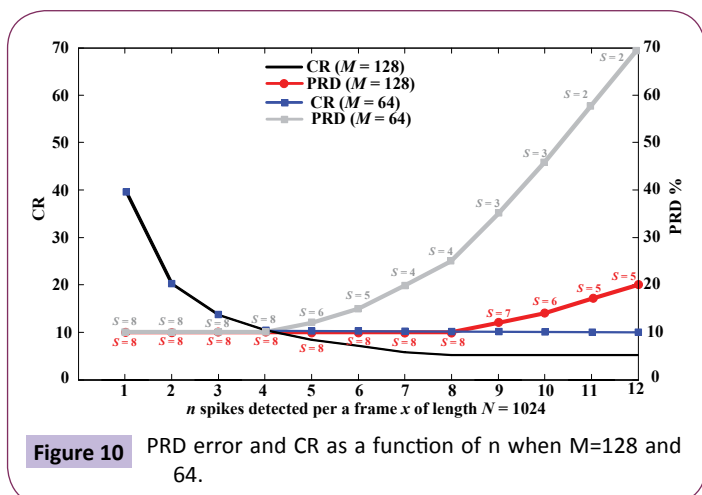


Figure 10 PRD error and CR as a function of n when $M=128$ and 64.

i. A spike can be detected on the boundary of two consecutive frames, F1 and F2 (e.g., out of the 32 samples constituting a spike, the first 10 samples might be in F1 and the last 22 in F2). Our solution consists in handling such a spike without compression (by allocating $S=32$). Therefore, instead of inserting a $J \times J$ wavelet matrix ψ_{si} in $\psi(2)$, we insert an identity matrix of size $((N - i) + 1) \times ((N - i) + 1)$ to recover F1, where i is the time of occurrence of the spike. Also, to recover F2, we insert an identity matrix of size $(31 - (N - i)) \times (31 - (N - i))$ at the starting point of ψ . This adjustment is triggered only when i of the spike in F1 is ≥ 994 .

ii. There are many neurons around electrodes, and they can fire simultaneously or one after another within a very short time. In these cases, spike waveforms from a few neurons overlap each other and the resulting spikes have distorted shapes. Hence, no prior knowledge of support for such spike can be learned. One solution consists in handling such a spike without compression (by allocating $S=32$ and inserting an identity matrix of size $J \times J$ in ψ). However, an appropriate detection of such spike needs to be integrated into the spike detection stage. This issue is currently under investigation. For now, the recovery of such spike can be done but with a relatively large PRD (because it is handled as a normal one).

Moreover, there is another issue regarding the spike detection technique (which is used to increase sparsity of the signal) before applying CS. In spike detection, we wish to decide between two hypotheses, the null hypothesis (i.e., only the noise is present) versus the alternative hypothesis (i.e., a spike and noise are both present). One needs to continuously compare the input signal to

a threshold to decide which hypothesis is true. The value of this threshold is generally based on estimation of statistical moments of the background noise [45]. To achieve an optimal trade-off between spike detection efficiency, circuit size, and low-power consumption, one can calculate the threshold on the receiver side (outside of the body) and then transmit it back to the implanted device. To perform this calculation, the neural signal needs to be transmitted without any processing. Therefore, the system must bypass the spike detector and the CS encoder altogether (**Figure 3**).

Finally, there is a great advantage in using CS-based methods in data reduction in wireless neural recording systems. Indeed, there is a decoupling between compression (encoder) and decompression (recovery) parts in CS-based methods. The CS encoder can be a 'generic' circuit while the decoder part can be adaptive, as in the recovery method presented in this paper. As a result, the signal recovery algorithm in the CS-based methods can be changed while keeping the same CS encoder. This is a powerful property which allows CS to accommodate evolution of technology and algorithms. For example, if our proposed adaptive recovery is improved or a new method is discovered in future, we can still ameliorate the data reduction capacity of the system without making change to the 'generic implanted device, obviating the need to perform a new surgical operation.

Acknowledgment

The authors would like to thank Dr. C. A. Chapman from Concordia University for providing recordings of neural signals.

References

- 1 Aghagolzadeh, Mehdi, Karim Oweiss (2009) Compressed and distributed sensing of neuronal activity for real time spike train decoding. *Neural Systems and Rehabilitation Engineering*, IEEE Transactions 17: 116-127.
- 2 Aghagolzadeh, Mehdi, Ali Mohebi, Karim G Oweiss (2014) Sorting and tracking neuronal spikes via simple thresholding. *Neural Systems and Rehabilitation Engineering*, IEEE Transactions 22: 858-869.
- 3 Kamboh AM, Mason A, Oweiss KG (2009) Resource constrained VLSI architecture for implantable neural data compression systems. *IEEE Symp. Circuit and Systems* 1481-1484.
- 4 Rehman S, Kamboh A (2014) A CMOS Micro-power and Area Efficient Neural Recording and Stimulation Front-End for Biomedical Applications. *Circuits Syst Signal Process* 1-22.
- 5 Shao R, Hu W, Wang Y (2014) The fault feature extraction and classification of gear using principal component analysis and kernel principal component analysis based on the wavelet packet transform. *Measurement* 54: 118-132.
- 6 Charbiwala Z, Karkare V, Gibson S, Markovic D, Srivastava MB (2011) Compressive sensing of neural action potentials using a learned union of supports. *IEEE Conf. Body Sensor Networks* 53-58.
- 7 Mamaghanian H, Khaled N, Atienza D, Vanderghyest P (2011) Compressed Sensing for Real-Time Energy-Efficient ECG Compression on Wireless Body Sensor Nodes. *IEEE Trans Biomed* 58: 2456-2466.
- 8 Chen F, Chandrakasan AP, Stojanović VM (2012) Design and Analysis of a Hardware-Efficient Compressed Sensing Architecture for Data Compression in Wireless Sensors. *IEEE Trans. Solid-State Circ* 47: 744-756.
- 9 Chen F, Chandrakasan AP, Stojanović V (2010) A signal-agnostic compressed sensing acquisition system for wireless and implantable sensors. *IEEE Conf. Custom Integrated Circuits* 1-4.
- 10 Calderbank R, Howard S, Jafarpour S (2010) Construction of a large class of deterministic sensing matrices that satisfy a statistical isometry property. *IEEE Sig Proc* 4: 358-374.
- 11 Amini A, Marvasti F (2011) Deterministic construction of binary, bipolar, and ternary compressed sensing matrices. *IEEE Trans Info Theory* 57: 2360-2370.
- 12 Colbourn CJ, Horsley D, McLean C (2011) Compressive sensing matrices and Hash families. *IEEE Trans Communicat* 59: 1840-1845.
- 13 Romberg J (2009) Compressive sensing by random convolution. *SIAM Journal on Imaging Sciences* 2: 1098-1128.
- 14 Tropp JA (2007) Random Filters for Compressive Sampling. *IEEE Conf. Informat. Scien and Syst* 216-217.
- 15 Chen X, Yu Z, Hoyos S, Sadler BM, Silva-Martinez Li J (2011) A sub-Nyquist rate sampling receiver exploiting compressive sensing. *IEEE Trans. Circ. Syst.* 8: 507-520.
- 16 Do TT, Gan L, Nguyenand NH, Tran TD (2012) Fast and efficient compressive sensing using structurally random matrices. *IEEE Trans. Sig. Proc* 60: 139-154.
- 17 Liu B, Zhang Z, Xu G, Fan H, Fu Q (2014) Energy efficient telemonitoring of physiological signals via compressed sensing: A fast algorithm and power consumption evaluation. *Biomedical Signal Processing and Control* 11:80-88.
- 18 Candès E, Wakin MB (2008) An introduction to compressive sampling. *IEEE Sig Proc Mag* 25: 21-30.
- 19 Cai TT, Wang L (2011) Orthogonal matching pursuit for sparse signal recovery with noise. *IEEE Trans Info Theory* 57: 4680-4688.
- 20 Chen SS, Donoho D, Saunders MA (1998) Atomic decomposition by basis pursuit. *SIAM J. of Scien Comput* 20: 33-61.
- 21 Efron B, Hastie T, Johnstone I, Tibshirani R (2004) Least angle regression. *The Annals of Statistics* 32: 407-99.
- 22 Murray JF, Kreutz-Delgado K (2001) An improved FOCUSS-based learning algorithm for solving sparse linear inverse problems. *IEEE Conf. Sig., Syst., and Comp* 347-351.
- 23 Wipf D, Rao BD (2004) Sparse Bayesian learning for basis selection. *IEEE Trans Sig Proc* 52: 2153-2164.
- 24 Mallat S, Zhang Z (1993) Matching pursuits with time-frequency dictionaries. *IEEE Trans Sig Proc* 41: 3397-3415.
- 25 Pati YC, Rezaifar R, Krishnaprasad PS (1993) Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. *IEEE Conf. Sig., Syst., and Comp* 40-44
- 26 Needell D, Tropp J (2009) CoSamp: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis* 26: 301-321.
- 27 Bankman IN, Johnson KO, Schneider W (1993) Optimal detection, classification, and superposition resolution in neural waveform recordings. *IEEE Trans Biomed Eng* 40: 836-841.
- 28 Obeid I, Wolf PD (2004) Evaluation of spike-detection algorithms for a brain machine interface application. *IEEE Trans Biomed Eng* 51: 905-911.
- 29 Drolet J, Semmaoui H, Sawan M (2011) Low-Power Energy-Based CMOS Digital Detector for Neural Recording Arrays. *IEEE Conference on Biomedical Circuits and Systems (BioCAS)* 13-16.
- 30 Eldar YC, Kuppinger P, Bolcskei H (2010) Compressed sensing for block-spars signals: uncertainty relations, coherence, and efficient recovery. *IEEE Trans Sig Proc* 58: 3042-3054.
- 31 Lv X, Wan C, Bi G (2010) Block orthogonal greedy algorithm for stable recovery of block-sparse signal representations. *Sig Proc* 90: 3265-3277.
- 32 Baraniuk RG, Cevher V, Duarte MF, Hegde C (2010) Model-based compressive sensing. *IEEE Trans. Info. Theory* 56: 1982-01.
- 33 Hegde C, Baraniuk RG (2011) Sampling and recovery of pulse streams. *IEEE Trans Sig Proc* 59: 1505-1517.
- 34 Ben-Haim Z, Eldar YC, Elad M (2010) Coherence-based performance guarantees for estimating a sparse vector under random noise. *IEEE Trans. Sig. Proc* 58: 5030-5043.
- 35 Huang J, Huang X, Metaxas D (2010) Learning with dynamic group sparsity. *IEEE Conf Comp Vision* 64-71.
- 36 Vidakovic B (1999) *Statistical modeling by wavelets*. A Wiley Interscience Publication. JOHN WILEY & SONS, INC. NewYork.
- 37 Chapa JO, Rao RM (2000) Algorithms for designing wavelets to match a specified signal. *IEEE Trans Sig Proc* 48: 3395-3406.
- 38 Gupta A, Joshi SD, Prasad S (2005) New approach for estimation of statistically matched wavelet. *IEEE Trans Sig Proc* 53: 1778-1793.
- 39 Kamavuako EN, Jensen W, Yoshidaa K, Kurstjens M, Farina D (2010) A criterion for signal-based selection of wavelets for denoising intrafascicular nerve recordings. *J Neurosci Meth* 186: 274-280.
- 40 Oweiss KG (2010) *Statistical signal processing for neuroscience and neurotechnology*, Academic Press, Elsevier.

- 41 The NeuroEngineering Lab of The University of Leicester, Leicester, Unitedkingdom.
- 42 Chronux Organization.
- 43 Hayes MH (1996) Statistical digital signal processing and modeling, NJ: John Wiley & Sons.
- 44 Quian Quiroga R, Nadasdy Z, Ben-Shaul Y (2004) Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. J Neural Comput 16: 1661-1687.
- 45 Semmaoui H, Drolet J, Lakhsassi A, Sawan M (2012) Setting adaptive spike detection threshold for smoothed-TEO based on robust statistics theory. IEEE Trans. Biomed Eng 59: 474-482.