An Improved SpikeTrack: An Autonomous Multi-Electrode Control & Recording System

Paul Hebert, Michael T. Wolf, Michael Campos, Shubhodeep Chakrabarti, Alexander Gail, Joel W. Burdick

Abstract—This paper summarizes an algorithm to autonomously position an extracellular recording electrode so as to first isolate the action potentials of a single neuron in a multiunit signal, and then re-position the electrode as necessary to optimize and maintain the recording quality of that neuron over an extended recording interval. We first summarize some of the technical advancements of the current algorithm over earlier versions of the “SpikeTrack” recording system in the area of multi-hypothesis cluster tracking method for spike sorting, and a new technique to optimize the signal recording interval. Novel recording experiments in macaque cortex compare the performance of autonomous extracellular recording with that of an experienced neurophysiologist. We found that the algorithm isolates cells better than a human expert.

I. INTRODUCTION

The need to obtain reliable, high quality neural electrophysiological recordings is a common problem in basic neuroscience as well as neuroprosthetics. Isolation of a specific neuron’s extracellular signal from background noise is a key requirement for subsequent scientific analysis or for decoding control signals in the case of a neuroprosthetic application. The isolation of a cell depends largely on how closely the electrode tip lies to the soma of the signal generating neuron. This paper describes a completely autonomous algorithm which can position extracellular recording electrodes so as to first isolate a neuron’s signal and then optimize and maintain that neuron’s signal quality over an extended recording interval. The work reported in this paper describes several significant improvements over a previous algorithm designed for autonomous recording electrode positioning [1]. In particular, we summarize an enhanced method to track neuronal signal clusters over time [2] and a new method to optimize the duration of signal recording. In addition, this paper presents novel recording experiments in macaque cortex which compare the algorithm’s ability with that of human neurophysiologist on the same recording tasks. In summary, we find that there is a slight statistical difference in the abilities of the autonomous positioning algorithm and an experienced neurophysiologist to isolate individual cells. Hence, automation of the acute extracellular recording process is a viable endeavor.

P. Hebert and J.W Burdick are with the California Institute of Technology, Pasadena, CA paul.hebert@caltech.edu
M. T. Wolf is with the Advanced Robotic Controls Group, Jet Propulsion Laboratory, Pasadena, CA
M. Campos is with the Dept. of Neurosurgery, Massachusetts General Hospital / Harvard, Boston, Mass.
S. Chakrabarti and A. Gail are with the Sensorimotor Group, German Primate Center, Gottingen, Germany.

II. ALGORITHM SUMMARY

The algorithm aims to autonomously position a recording electrode within a brain region of interest so as to first isolate an individual neuron, and then reposition the electrode as necessary so as to maintain high signal quality over an extended recording interval. Figure 1 shows a simplified view of the main control cycle which governs the motion of a single electrode (for multiple electrodes, multiple instances of the algorithm control each electrode), with cycle iterations indexed by \( k \) \((k = 1, 2, \ldots)\).

The control cycle begins by recording voltage from the stationary electrode over a time interval \( T_k \). Action potential, or “spike,” waveforms are detected within this interval using a wavelet-based approach [3]. A short fixed length snippet of the raw signal surrounding the detected action potential waveform (the \( i^{th} \) spike waveform detected in \( T_k \) is denoted \( s_{i,k} \)) is extracted and all spikes found in \( T_k \) are aligned along their minimum peaks. The waveform snippets are then projected to a 2-dimensional feature space using Principal Component Analysis (PCA) (note that other feature spaces can be used with our approach). These projected waveforms, \( x_{i,k} \), are now used as the basis of a cluster-based spike sorting process. As described in the next section, we use a novel cluster based multiple-hypothesis tracking and clustering algorithm to sort the spikes and to maintain multiple probabilistic hypotheses about all signal histories. Signal quality metrics are calculated for each cluster, and these
results are used to determine the optimal electrode movement needed to maintain the dominant neuron’s signal quality. A finite state machine control system manages the electrode’s movement to the optimal depth, while also monitoring and accounting for a variety of issues such as maximum electrode speed, the possibility of neuron impalement, etc.

III. REVIEW OF IMPROVEMENTS

The basic architecture shown in Fig. 1 has already been established in our earlier work [4]–[7]. This section summarizes the major enhancements made to key portions of the overall system, which result in improved robustness and performance.

A. Clustering

Reliable clustering of spike data and tracking of the neural signal sources over time is critical for calculating metrics of neuronal clusters, which are the basis for control decisions. Corruption in the sorting process may lead to poor algorithm performance, and subsequently to non-optimum recording and degraded quality of the subsequent interpretation of the recordings.

Because an autonomous recording electrode may continually move, a spike sorting method compatible with this characteristic must be able to associate each cluster \( C_g \) found in recording interval \( T_k \) with neuronal signal clusters found in previous intervals \( (T_{k-1}, T_{k-2}, \ldots) \) in order to “track” individual neuronal signal sources over time. Because the data association across recording intervals can be uncertain, and because neurons can become silent, the algorithm must also maintain a strict probabilistic interpretation of all signal histories.

The spike sorting technique summarized in this paper builds upon the unsupervised Gaussian Mixture Model (GMM) clustering method of Wolf and Burdick [8]. The distribution of each cluster, \( C_g \), in feature space is modeled as a generating gaussian distribution \( f_{\Theta}(X_k | \mu_g, \Sigma_g) \). Assuming independence of measurements, the likelihood of the model is

\[
p(X_k | \Theta_k) = \prod_{i=1}^{N} \sum_{g=1}^{G_k} \pi_{g,k} f_{\Theta}(X_k | \mu_g, \Sigma_g),
\]

where \( X_k = \{x_{i,k}\}_{i=1}^{N} \) is the collection of all waveforms in \( T_k \), the mixture model parameters \( \mu_g, \Sigma_g \) are the mean and covariance of the \( g \)th cluster found in interval \( T_k \), and \( \Theta_k \) denotes all mixture model parameters at interval \( T_k \). These parameters, the distribution of the clusters (Gaussian), and the model order number, \( G \), form the mixture model class, \( M \).

Clustering in the GMM framework is carried out using Expectation-Maximization (EM) in order to maximum the likelihood (1). Improving upon this, instead of using EM on the data likelihood, Wolf & Burdick [8] apply EM to the posterior:

\[
p(\Theta_k | X_{1:k}) \propto p(X_k | \Theta_k) p(\Theta_k | X_{1:k-1}),
\]

in order to properly incorporate clustering results from previous recording intervals. The prior is naturally constructed from the mean cluster location \( \mu_{g,k} \), since the other model parameters \( \Sigma_g \) and \( \pi_{g,k} \) are too variable from one interval to another to provide a meaningful prior. Thus, the prior can be formed as

\[
p(\Theta_k | X_{1:k-1}) = \prod_{g=1}^{G_k} \left( \sum_{j=1}^{G_k-1} \frac{1}{c_k-7} N(\mu_{g,k}, \psi_{j}^{k}) \right)
\]

where \( \psi_{j}^{k} \) is comprised of estimated mean \( \bar{\mu}_{j,k-1} \) and the covariance \( S_{j,k-1} \) associated with that estimate. In addition to this clustering methodology we add a “tracking” ability which is described in the next section.

B. Tracking

To robustly “track” neurons over time (i.e., to robustly identify the specific neuron which generates a given cluster of waveforms), we have extended the multiple hypothesis tracking framework of Reid [9] (and the computational management of such hypotheses from Murty [10]) which generates the L-best hypotheses) to the case of clustered data measurements. The basic idea is to generate and maintain hypotheses about how data in one recording interval is related to data in a previous recording interval (recalling that the electrode may move between intervals). Unlike the classical multiple hypothesis tracking approach, which concerns only about one level of association between a current measurement and a target “track” synthesized over time from previous measurements, the neural tracking problem requires the solution to a two-level data association problem. First, each spike detected in \( T_k \) must be associated to one cluster. There may be multiple plausible ways in which the data recorded during \( T_k \) can be clustered, with each of these possibilities termed a model hypotheses, denoted by \( M_t = \{M_t(m)\} \) for the \( m \)th model in \( T_k \). A data association hypothesis, denoted by \( h_l = \{\tau_l, \nu_l, \phi_l\} \), assigns each putative neuron (as represented by a cluster in \( M_t(m) \)) found in \( T_k \) to either the “track” of a neuron identified in previous recording intervals (\( \tau_l \) is the set of neurons being tracked), or to a set \( \nu_l \) of clusters for newly appeared neurons, or to \( \phi_l \), the set of clusters that are deemed spurious or false. Note that the cardinalities of each set \( -N_r, N_c, N_o \) sum to the total number of clusters in the given clustering model (i.e. \( G = N_r + N_c + N_o \)) and \( l \) indexes the \( L \) hypotheses.

![Fig. 2: Multi-hypothesis tracking tree for L=4.](image)
hypothesis: it maintains the complete set of data associations, which includes spike waveform features-to-cluster associations $M_{i}(t)$ and the cluster-to-neuron associations $h_{l}$ during each interval $T_{k}$. The Global Hypotheses, $H^{1:k}_{i} = \{H^{1:1}_{i}, H^{1:k}_{i} \}$, combine the the joint hypotheses with all parent joint hypotheses $H^{p(k)}_{i}$ to provide the history of the current L-best joint hypotheses. Figure 2 illustrates the generation and pruning of the model and data associations for the case of $L = 4$. The key computation for MHTC is the global hypothesis probability, which provides the basis for ranking the final hypothesis and selecting the best one at time $k$. Very briefly, the probability can be expressed using a combination of Bayes’ rule, the chain rule and Laplace’s method:

$$P(H^{1:k}_{i} \mid X^{1:k}) \approx \frac{1}{C} \sum_{n \in \Gamma} \prod_{(g,j) \in \tilde{h}_{n}} a_{gj} \quad \rho \quad \sum_{l} P_{3} P_{4} P_{5}, \quad (2)$$

where $a_{gj}$ are the likelihoods of assigning the $g^{th}$ cluster to the $j^{th}$ neuron/target, $P_{3}$ is the model hypothesis prior, $P_{4}$ is the model evidence, $P_{5}$ is the probability of the parent hypothesis and finally $C = p(X^{k} \mid X^{1:k-1})$ is a constant and is the same for any hypothesis. Detailed formulas for computing these comprising probabilities as a function of the recorded data, as well as implementation details on the multiple hypothesis tracking for clusters (MHTC) algorithm, can be found in [2].

C. Calculating Metrics

Our algorithm tries to simultaneously optimize each neuron’s isolation and signal within all current joint hypotheses $H^{1:k}_{i}$ in interval $T_{k}$. We calculate a neuron’s isolation by evaluating the Minimum Average Mahalanobis Distance (MAM)

$$\text{MAM}_{d} = \min_{i \in \{1,...,G\}; i \neq d} \left\{ \frac{1}{2} \sum_{g \neq i} \sum_{i \neq d} \left( \mu_{i} - \mu_{d} \right) \Sigma_{g}(\mu_{i} - \mu_{d})^{T} \right\} \quad (3)$$

between the given neuron and other detected neurons. MAM provides a better discriminability metric for isolation by incorporating the interaction of both clusters through the use of their covariances.

The Signal-to-Noise ratio (SNR) is used to characterize signal quality. This metric is calculated on the collection of waveforms $X^{g}_{k}$ associated with cluster $C_{g}$, using their Peak-to-Peak (PTP) amplitudes:

$$\text{SNR}_{g}^{k} = \frac{1}{N} \sum_{i=1}^{N} \frac{PTP_{i}}{V_{\text{Noise RMS}}} \quad (4)$$

where $V_{\text{Noise RMS}}$ is the RMS value of noise samples collected from the current interval and $N$ is the number collected waveforms in $X^{g}_{k}$.

As the MHTC algorithm track individual neurons from interval to interval, a history of a target’s SNR can be built. The goal is to optimize and isolate the dominant target/neuron - the neuron with the current highest SNR in the history. Unlike our previous algorithm, the goal is to now optimize and isolate the dominant target neuron in the leading global hypothesis, $H^{1:k}_{i}$.

D. Optimization & Isolation Control Loop

In order to isolate the leading global hypothesis’ dominant neuron and optimize it’s signal, we implement a finite state machine which controls the state of the electrode movement process. Our supervisory finite state machine (SFSM) as shown in figure 3, uses the current state of the algorithm, current and past calculated metrics to determine the appropriate movement of the electrode. We describe below the prototypical pathway of the SFSM which attempts to isolate and maintain isolation of a neuron. At the outset, an electrode is presumably in an area of interest within the brain. The algorithm starts off in Spike Search mode which attempts to discover an active area of the brain. The electrode will continue to move in $\Delta_{\text{search}}$ increments until a sufficient amount of spikes is collected, both of which are pre-set by the experimenter. With sufficient spikes and minimal signal & isolation metrics, the algorithm then switches into Gradient Search mode in an attempt to build a gradient in the SNR vs. position curve. The step size in this mode until a gradient is found is $\Delta_{\text{sample}}$. After the initial move, the optimization procedure described in [7] and [4] determines the best move. If the optimized regression determines $n_{1}^{*} = 1$ ($n^{*}$ is the number of basis functions used in the regression, hence the best fit curve is a horizontal line), the electrode will continue to move at $\Delta_{\text{sample}}$. However, if $n_{1}^{*} > 1$, signaling an optimizable SNR vs. position curve has been found, the state of the SFSM switches to Optimize Signal, and while in this state, the move command is determined from the optimization procedure. Once a maximum is found on the curve, the state then moves into Neuron Isolated. If the number of spikes or either signal or isolation metrics fall below the requirements, the SFSM will enter a similar re-isolation process. Figure 3 also details other corner cases.

IV. Experimental Results

An implementation of an earlier version of the autonomous electrode positioning algorithm logged over a 1,000 hours of recording time. Figure 4 shows a recording session that demonstrates the clustering and tracking ability. To assess the
The impact of the improvements summarized in this paper, and to also begin to assess the practical potential for autonomous electrode positioning algorithms, the algorithm summarized above was experimentally evaluated in macaque parietal cortex.

The experiment compared a human expert’s performance of isolating cells versus SpikeTrack’s. Electrodes were lowered into the dorsal premotor cortex of an awake behaving rhesus monkey and allowed to settle for 30 minutes. Following this, neural activity was recorded for a variable length of time. Re-positioning of electrodes to maintain the neurons in an isolated state was achieved either by a human observer or by SpikeTrack. Recorded data from both the manual and SpikeTrack recordings were sorted-blind offline by an experimenter. Briefly, isolated clusters in PCA space were used to define single units. Only single units that were maintained in an isolated state for a minimum of 100 seconds were included in the analysis.

Each neuron’s total isolation time was calculated from the difference between the first and last timestamps of that neuron shown in figure 5. Total file time was defined as the difference between the first isolated timestamp and the last isolated timestamp across channels. A histogram of the absolute total neuronal isolation times for files recorded using SpikeTrack is shown in figure 6 (mean = 3246 sec.). Since both the total number and the duration of the manual and SpikeTrack files were different, files were randomly chosen a 1000 times from the manual set in order to match (in number and duration) the SpikeTrack files. For each shuffle, a mean relative isolation time was calculated. Figure 7 shows the average mean isolation times along with their standard deviations. The $\alpha$ value indicates SpikeTrack obtained longer isolation times than a human experimenter in 991 out of 1000 shuffles.

References