



Master's thesis project: Sensorimotor integration and learning of artificial stimuli

Laboratory: Sabes Lab, Center for Integrative Neuroscience, UCSF

Lab Director: Philip N. Sabes, Principal Investigator

Lab Supervisor: Joseph E. O'Doherty, Postdoctoral Scholar

November 2016 to Mai 2017

AVALOS Diana

Contents

1	Introduction	4
2	Presentation of Philip Sabes Laboratory	5
3	Non-primate experiments	6
3.1	Trainings	6
3.2	Procedure	6
3.2.1	Preparation	6
3.2.2	Running the experiment	6
3.2.3	End of the experiment	7
4	Methods	8
4.1	Subject and Implants	8
4.2	Recording and stimulation protocols	9
4.3	Spike sorting	10
4.4	Hand Reaches	13
4.5	Artifact Removal	14
4.6	BMI task	15
4.7	Analysis of the performance at the BMI task	16
4.8	Detection task	17
5	Evolution of stimulation sensitivity across time	18
5.1	Introduction	18
5.2	Methods	18
5.2.1	Monkey's task	18
5.2.2	Quest algorithm	18
5.2.3	Results	18
6	Implementation of a Spike Sorting Algorithm	20
6.1	Neural Signals and Electrophysiological Recordings	20
6.2	Requirements for the design of the algorithm	20
6.3	Challenges in Spike sorting	21
6.4	Pre-processing of the data	21
6.5	Feature Selection and Dimensionality Reduction	21
6.6	Clustering	24
6.6.1	Gaussian Mixture Model	24
6.6.2	Gaussian Mixture Model with Isosplit	26
6.6.3	Density-based spatial clustering algorithm	28
6.7	Further Perspectives in Spike sorting	31
7	Craniotomy surgery	33
7.1	Preoperative animal preparation and Anesthesia techniques	33
7.2	Intraoperative monitoring	33
7.3	Implantation techniques	33
7.4	Postoperative care	34
8	Conclusion	36
9	Acknowledgements	36
10	Bibliography	36

List of Figures

1	Map of brain regions in primates	8
2	Cortical homonculus	9
3	Microelectrode array	9
4	Stimulation array	10
5	Spike sorting software	11
6	Spike sorting of four channels	11
7	Spikes alignment and PCA of one channel	12
8	Broadband of 4 channels	13
9	Experimental set-up	14
10	Screenshot of the reaching task.	14
11	Screenshot of the screen of the human operator monitoring the task	16
12	Results of the experiment	17
13	Results of the Detection task	19
14	Comparison of 2 different methods of dimensionality reduction	23
15	Spikes extracted from the GMM algo from 10 and 3 Principal Components	24
16	Akaike and Bayesian Information Criterion to choose the number of clusters of the model	25
17	Akaike Information Criterion to choose the number of clusters of the model	26
18	Computation of the Hartigan's DIP statistic from a range of Gaussian distributions	27
19	Variability of the GMM and Isosplit algorithm	27
20	Comparison of the GMM with 4 clusters and GMM Isosplit with 10 clusters	28
21	DBSCAN algorithm scheme	29
22	DBSCAN algorithm: tuning the parameters	30
23	DBSCAN algorithm: clusters selected	30
24	DBSCAN algorithm: waveforms of the selected clusters	31
25	Areas targeted for the electrode arrays (one in M1, the other in S1).	34
26	Mapping of the electrodes receptive fields.	35

1 Introduction

This report presents the work accomplished during my six months internship in my 3rd year of Biomedical Engineering at the Grenoble Institute of Technology Phelma (National Graduate Engineering School). This internship was carried out at Sabes Lab, in the Center for Integrative Neuroscience, which belongs to the University of California, San Francisco (UCSF).

I specifically wanted to realize my master thesis project in a laboratory working on neural prostheses to help restore sensory, motor and cognitive function for people suffering from disabling neurological conditions. Indeed, my exchange year at EPFL developed my curiosity in neuroprosthetics, through different courses and group projects. More specifically, I was captivated by the project developed in the scope of the brain-computer interaction course with Prof. J. R. Milln, which aimed at developing a real-time control algorithm of an exoskeleton hand from electroencephalography recordings. Another important project which developed my curiosity in neuroprostheses is the one developed during the sensorimotor neuroprosthetics course with Prof. G. Courtine and Prof. S. Micera, which consisted in inventing a new neuroprosthesis: we proposed a bionic glove combined with peripheral nerve stimulation for somatosensory senses restoration and an artificial muscular reflex of wrist extension to avoid dangerous situations for serious burn victims. These experiences were inspirational for me and reinforced my motivation for a Master Thesis Project in the field of neuroprosthetics.

I applied to Philip Sabes laboratory after reading a paper published by his lab in 2015, titled: *A learning-based approach to artificial sensory feedback leads to optimal integration*. Indeed, the lab focuses its research on sensorimotor integration and learning, and artificial feedback for patients using BCI. I worked with Joseph E. O'Doherty and Mariana Cardoso (Postdoctoral Scholars) on the next steps of this study. The project is about having a monkey able to use a bidirectional brain-control interface, controlling the cursor's movement with his motor cortex and integrating an artificial proprioceptive feedback on the cursor's position and speed, delivered in his somatosensory cortex through intra-cortical electrodes. The paper mentioned above achieved to show that combination of an initially unfamiliar artificial sensory signal with vision leads to greater accuracy and speed in goal directed movements. The next step is to degrade the visual feedback, and show that the monkey can almost completely rely on the artificial feedback only.

In this report, I explain in details the project and my contribution, as well as the experience I have acquired during this internship. I contributed in several domains: I assisted with brain surgery on non-human primates for the implantation of electrode arrays in the sensory and motor cortices. I ran non-human primate experiments using multichannel intracortical microstimulation in goal-directed movement tasks and improved the spike-sorting algorithm used in the lab for brain computer interaction.

2 Presentation of Philip Sabes Laboratory

The laboratory uses a combination of cortical physiology (study of mechanisms and interactions which operate within the cortex), psychophysics (relationship between physical stimuli and the sensations and perceptions they produce), and computational modeling to study the cortical mechanisms for sensorimotor integration and learning. It focuses on discovering the physiological mechanisms and computational principles by which information is processed in cortical sensorimotor circuits and how sensorimotor experience continually reshapes these circuits.

Among the main techniques used by the lab regarding cortical physiology, there are large-scale intra cortical recordings, which enable to simultaneously record from large neural populations and perform quantitative analyses at the level of the population responses in multiple brain areas [1]. The lab also manipulates cortical activity to measure changes in network dynamics and drive those changes by directly manipulating patterns of cortical activity. Two approaches are being developed: patterned electrical stimulation and patterned light stimulation in tissue expressing light-sensitive ion channels (commonly called optogenetics) [2]. Furthermore, computational and theoretical models are used to link our understanding of brain and behavior [3-4]. These models generate testable hypotheses about the dynamics of cortical networks, and we use these models to design the next physiological experiments.

Proprioception, which is the sense of the body's position in space, is essential to natural movement planning and execution. [5-6]. Furthermore, several studies showed that more precise movements are achieved by combining estimates of multiple sensory modalities [7-10], for instance vision and proprioception. Thus, designing successful motor prostheses and brain-machine interfaces (BMIs) with naturalistic control will require artificial proprioception, potentially delivered via intracortical microstimulation (ICMS).

Several studies in the lab investigated how the brain decodes ICMS and integrates multimodal sensory signals. The lab has previously shown that ICMS can guide a monkey performing reaches to unseen targets in a planar workspace [11], and studied how ICMS is decoded into target angle and distance by analyzing the performance of a monkey when ICMS feedback was degraded [12]. More recently, another paper published by the lab [1] demonstrated that a learning-based approach can be used to provide a rich artificial sensory feedback signal. ICMS was providing continuous information about hand position relative to an unseen target. In this study, monkeys combined ICMS with vision, thanks to the sensorimotor learning and plasticity abilities of the brain, to form an optimal estimate of relative hand position and complete the reaches. During my internship in the lab, I helped for the next step of this project, which was encoding the position and speed of the hand position through ICMS (a sort of artificial proprioception) and degrade the visual feedback. Results show that the monkey is able to understand the artificial feedback and use it to improve its performance when performing goal-directed movements.

3 Non-primate experiments

3.1 Trainings

My work on the lab involved non-human primate experiments. The first weeks in the lab were dedicated to several trainings: Biosafety, Basic Regulatory and Ethical Requirement, Controlled Substances Safety, Laboratory Safety for Researchers, Annual Safety Training, Herpes B, Non Human Primates training.

Non-human primates can carry the Herpes B virus. In the natural host this virus is asymptotic but in humans this virus can cause severe disease (and treatment is not completely effective). Only 3% of the macaques carry this virus, but they can be tested positive or negative at different moments. Thus, we assume that they are all potentially infectious, and follow very strict safety rules. The training courses covered explanations about the Herpes B virus, disease in humans, transmission, and post-exposure procedures. Other training courses focused on prevention and safety rules to apply when running experiments with non-human primates. Indeed, one needs to get protected from bites and scratches incurred from macaques or injuries from contamination with macaque secretions. I also received a non-human primates training, explaining how to interpret macaque behavior, how to interact with them, and the potential diseases they carry.

3.2 Procedure

One of my tasks in the lab was training the monkey on the experiment designed for the study (as explained in the following chapter). This training was performed every day, since the monkey needs time and repetition to learn and succeed at the designed task. Also, the experiments have to be reproduced many times in order to gather a significant amount of data and obtain statistical results. Here, I explain my every day process of monkey training.

To prevent any contamination from the monkeys, the operator has to wear the Personal Protective Equipment while running the experiment (2 pairs of gloves, bouffant cap, procedure mask, lab clothes with a lab coat on top, face shield, special shoes with shoe covers).

3.2.1 Preparation

First, the monkey is taken out of his cage and installed in his special chair. After weighting him, the headpost is fixed to keep his head still; the headstages are connected to the 2 connectors on his head. The monkey is dressed and an electromagnetic sensor is attached to his left arm, under the sleeve. This enables to keep track of his hand position during the task. The liquid reward system is filled with juice and fixed to the chair, so the monkey gets a liquid reward when he performs the task well. The headstages are connected to the amplifiers, which are connected to the computer. The monkey chair is fixed to the table where the experiment is performed. The set up of the experiment is presented in Fig. 9.

3.2.2 Running the experiment

On the computer, directories are set up, and the program is launched to enable the data stream from the processor to the computer. The neural pipeline and the custom spike-sorting program are started. All channels are controlled (update of the threshold, noisy channels disabled, spike sort until satisfied). The first recording is the Quiet Sitting Task: five minutes of extracellular signals are saved, in which the monkey isn't actually performing any task. Afterwards, the stimulator is plugged into the sensory cortex connector and the software retrieving information about the position of the electromagnetic sensor on the monkey hand is activated. Spike sorting is performed at everytime, since there is enough alternation of the neural signal between two training sessions.

The monkey performs the hand-reaches task, to correlate neural activity with the intention of movement. Then, the stimulation pulse scheduler is started to train the Artifact Subtraction Filters, which reduces the amplitude of undesired transient voltages in the recordings. Finally, the brain-machine interface task is launched from Matlab, and the monkey performs goal directed reaches by controlling the neural cursor that has just been trained. At the end of the experiment, all the programs are shut down and the data is backed up on a server.

3.2.3 End of the experiment

After the experiment, the wound around the monkey's headpost needs to be cleaned and treated with antibiotics. Indeed, the immune system identifies the headpost and the headstages as foreign bodies and the wound around them is permanent. The monkey is brought back to his cage, and receives food and water. The monkey chair and the room are cleaned.

4 Methods

4.1 Subject and Implants

The experiments were conducted in one adult male Rhesus macaque monkey. Two 96 channel-silicon microelectrode arrays (Blackrock Microsystems, Salt Lake City, UT) (Figure 3) were chronically implanted into the right sensorimotor cortex (July, 2016) (Figure 1). Each array consisted of 96, 1.0mm long electrodes, spaced at 400-m and covering a 4 mm squared area. One array implanted over the primary somatosensory cortex (Brodmann area 1, S1) was used to deliver stimulation and the one implanted over the primary motor cortex (Brodmann area 4, M1) was used for recordings. The areas targeted for implantation have receptive-fields spanning in the arm and shoulder (Figure 2) If we target the right arm, the electrodes have to be implanted in the left hemisphere of the brain.

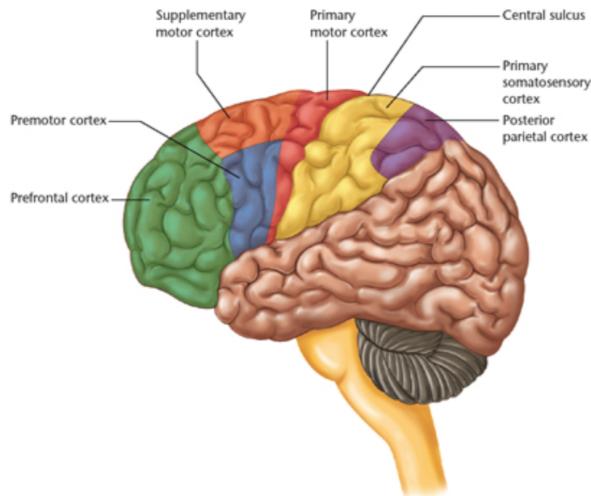


Figure 1 – Map of brain regions in primates. Microelectrode arrays were implanted into the right sensorimotor cortex. [13]

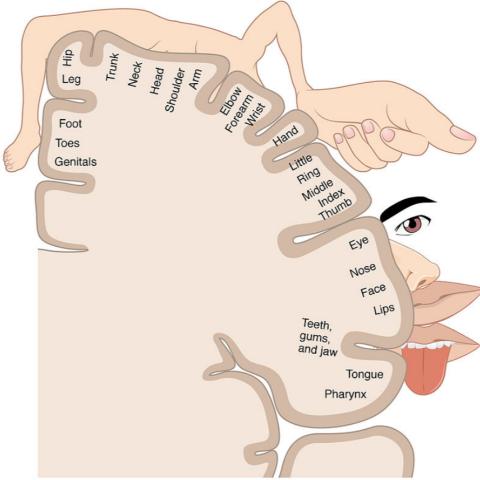


Figure 2 – Cortical homunculus: neurological ”map” of the anatomical divisions of the body. There are two types of cortical homunculi: sensory and motor. Primates and humans share a similar brain morphology. This map helps targeting the appropriate receptive field of the arm when implanting the electrodes. [14]



Figure 3 – 96 channel-silicon microelectrode array [15]

4.2 Recording and stimulation protocols

Recordings were made using RZ2 BioAmp Processor with PZ2 Preamplifier (Tucker- Davis Technologies, Alachua, FL) at a sample rate of 24.4kHz, which implies a temporal resolution for recorded waveforms of approximately $2.6 \mu\text{s}$ for 64 samples. A computer- controlled stimulator (IZ2, Tucker-Davis Technologies, Alachua, FL) was delivering electrical stimulation through symmetric, biphasic, cathode leading, charged balance, square-wave waveforms. Electrical stimulation was only delivered to the array implanted in the primary somatosensory cortex. For each of the 16 pairs of electrodes selected in this array, current amplitude was set between 20 and $75 \mu\text{A}$, at the level at which the monkey could perform the ICMS detection task with 75% accuracy, implemented via a two-target forced choice protocol. Stimulation currents are kept under $75 \mu\text{A}$ for safety reasons.

Stimulation doesn't try to recreate patterns of neural activity underlying natural sensation. Instead, it relies on the sensorimotor learning and plasticity of the brain. In previous studies, the lab demonstrated [1] that spatiotemporal correlation between a visual feedback and a new artificial feedback is sufficient for the monkey to integrate the new sensory modality. When stimulation is delivered, it takes the form of a patterned multichannel pulse train. 8 pairs of electrodes were encoding for the position of the cursor. Each one was assigned one of the 8 preferred directions, which were equally spaced around a circle. 8 other pairs of electrodes encoded the velocity of the cursor. Positions of the electrodes on the stimulation array are shown in Fig. 4.

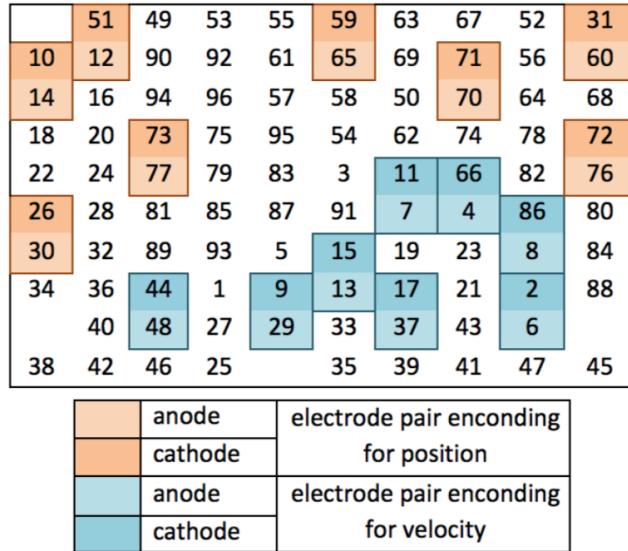


Figure 4 – Structure of the stimulation array

Note: pair 31-60 replaced by 52-56 at some point because loss of sensitivity around these electrodes.

4.3 Spike sorting

Neuronal activity (i.e. the electrophysiological signal) is defined by the firing of action potentials, which are electrical impulses emitted by neurons, also referred to as spikes. Spike sorting is the process of separating the electrophysiological signal into groups of single-unit activity corresponding to single neurons. Indeed, each electrodes record signals from a group of neurons and we need to attribute to each neurons its individual spikes, thanks to the definition of its characteristical shape. It is further explained in the spike sorting chapter. Spike sorting is realized with the custom made spike-sorting software of the lab. The neural signal visualized in the interface has already been filtered with a linear phase high pass FIR filter at 300Hz (to remove the DC component without phase distortion) and with an IIR low pass filter at 5kH (to remove high frequency noise).

The first step is to detect the noisy channels and disable them (so they are not taken into account for the experiment). Then, the threshold value is adjusted for all channels, usually at 3.95 times the standard deviation (Fig. 5). When the signal crosses the threshold, a spike is detected. The detected events are aligned with their minima. The extracted waveforms have 64 data points or features sampled at 24.4kHz, which means they last $2.6\mu\text{s}$. The 2 main principal components extracted from the raw spike data are projected in a 2 dimensional space to let the manual user sort the neurons by drawing cluster boundaries with the help of the user interface (Fig. 6-7).

We notice in Fig. 7 different spike shapes assigned to different neurons. The variability in spike shapes is due to the type of neuron, and its distance and relative position to the recording electrode.

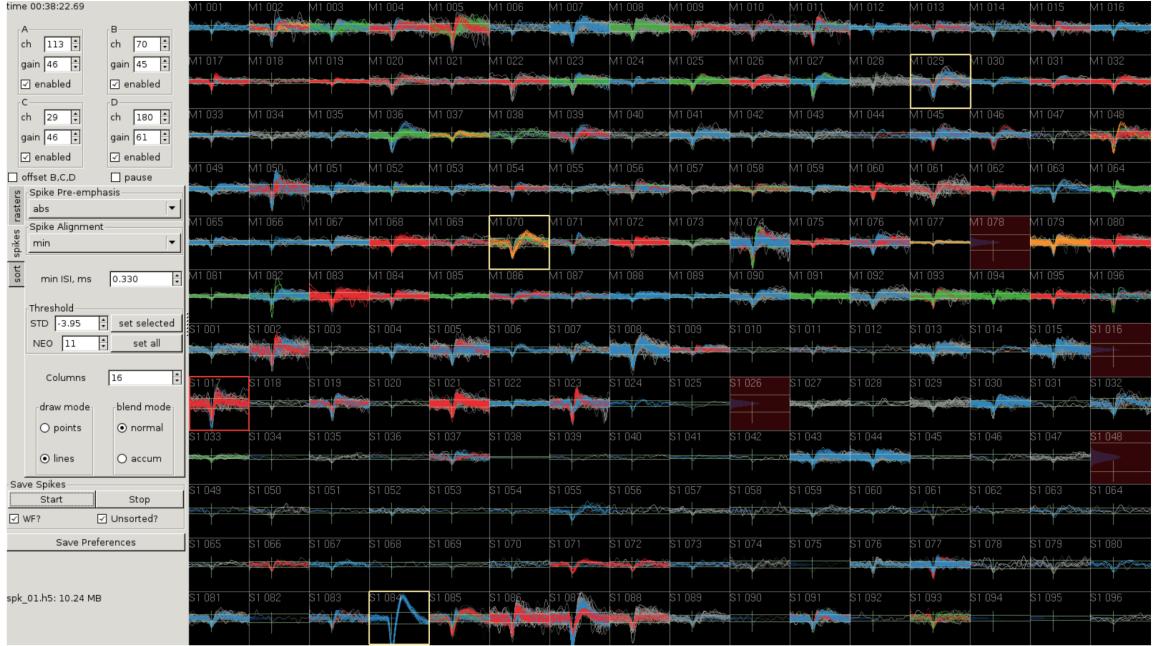


Figure 5 – Screenshot of the spike sorting software. All channels from both arrays are displayed here, and we can visualize their spiking activity. The first ones with a name starting with M belong to the array located in the motor cortex, and the other ones starting with S are located into the sensory cortex. A red background means that the channel was too noisy and has been disabled. On the left, we can see the parameters of the spike-sorting algorithm: alignment of the spikes to their minima, threshold set at 3.95 times the standard deviation, etc.

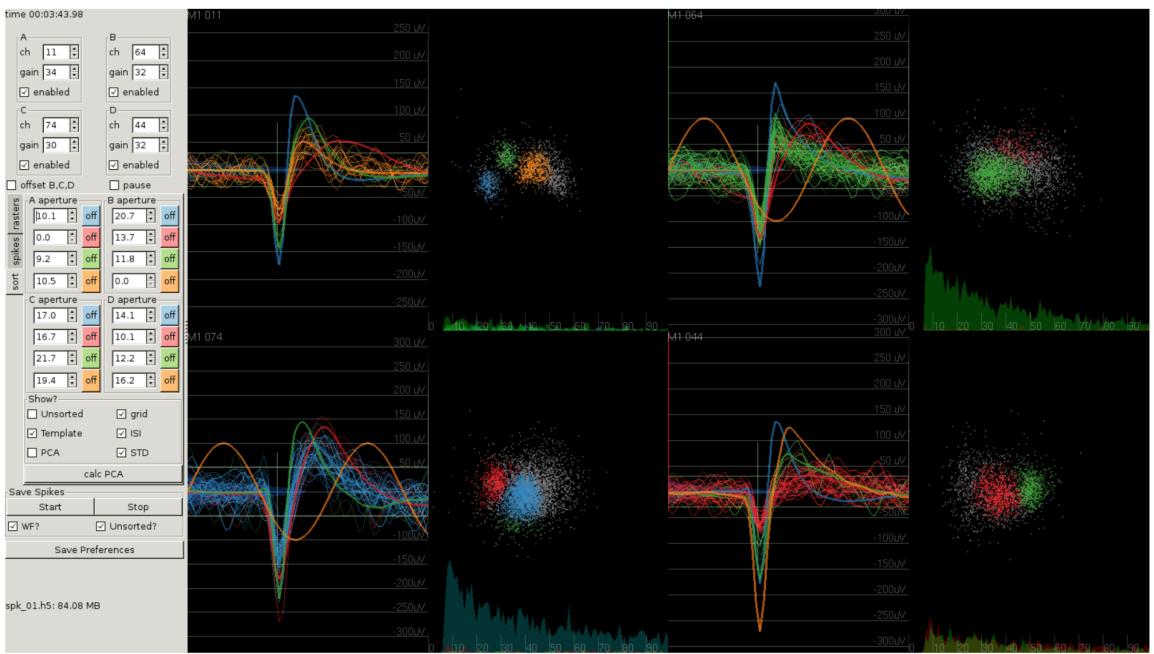


Figure 6 – Screenshot of the spike sorting software. 4 channels are displayed here, to get a better view of the spikes and enable manual spike-sorting of the clusters.

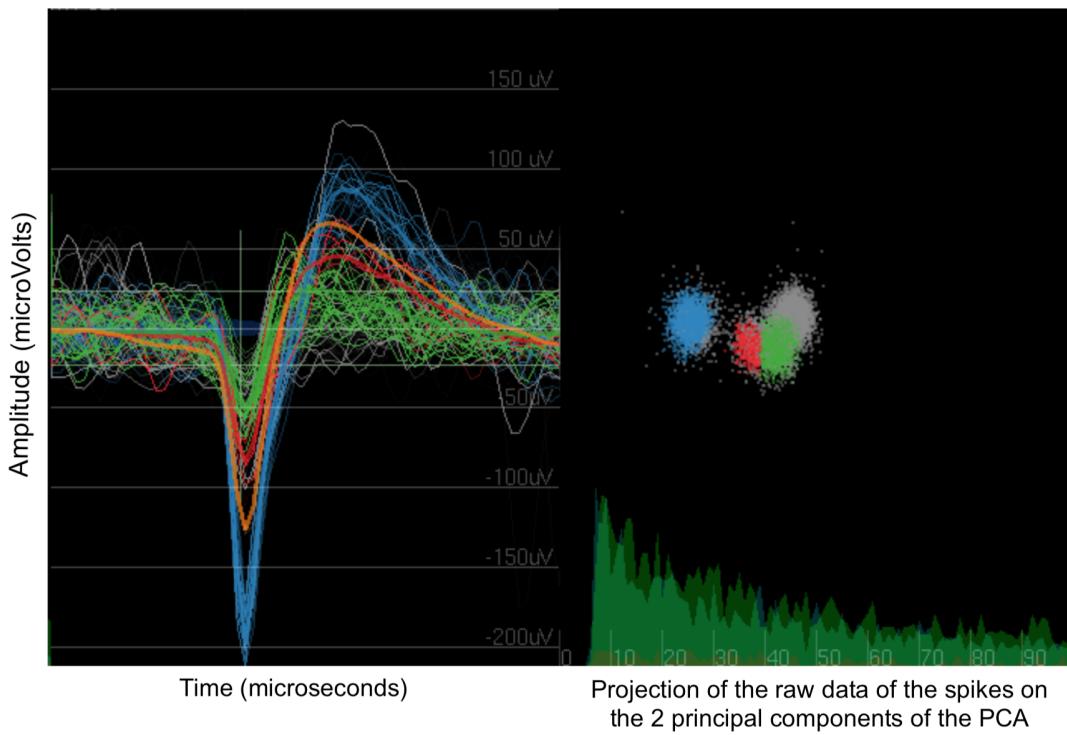


Figure 7 – Spike sorting of one channel. In the left, we observe the detected spikes and the color corresponding to their assigned cluster. On the right, the spikes are projected on the 2 first principal components. Each detected spike is represented by a point. By selecting a specific point with the mouse, its spike shape is highlighted on the right. The clusters are defined manually by selecting areas and assigning them a color (to differentiate the clusters).

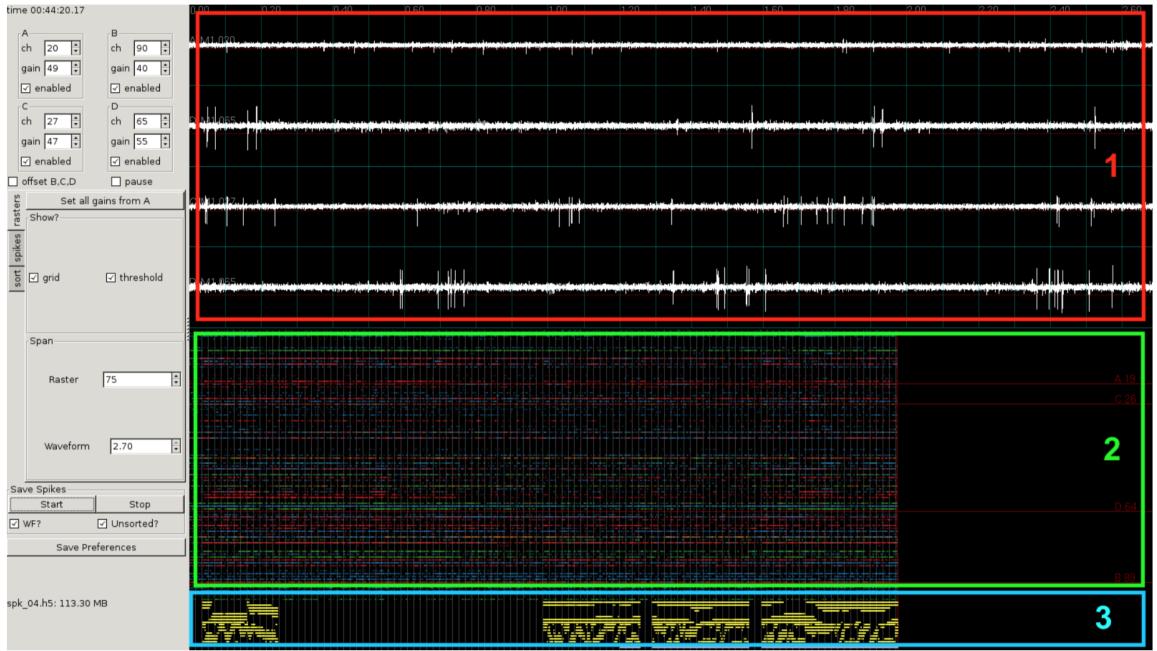


Figure 8 – Another screenshot of the spike sorting interface. In area 1, we observe the broadband of the 4 selected channels, after band-pass filtering (300Hz-5kHz). We notice the spikes standing out of the signal. In area 2, each horizontal line corresponds to the evolution of one channel over time. Color points represent the neural activity. In area 3 is represented the stimulation activity. The first 8 horizontal lines represent the activity of the 8 electrode pairs encoding for position, and the 8 following ones represent the activity of the electrode pairs encoding for velocity.

4.4 Hand Reaches

In this task, the primate was performing reaches to specific targets in the horizontal plane in a two dimensional virtual environment, where a mirror and an opaque barrier prevented direct vision of the arm. The mirror reflected images coming from visual output of the projector, and the visual input appeared to be in the horizontal plane of the reaching hand. Hand position was monitored using an electromagnetic tracker (Polhemus Liberty, Colchester, VT) placed on the monkey's hand. The monkey received visual feedback about the position of his hand in the form of a white, 10mm diameter circle. Targets appeared as green, 10mm diameter circles. A scheme of the set up of the experiment is provided in Fig. 9. and a screenshot of the reaching task is shown if Fig. 10. This hand-reaching task enables the algorithm to correlate the neural activity with the intention of movement (through the use of a Kalman filter), which is used in the BMI task.

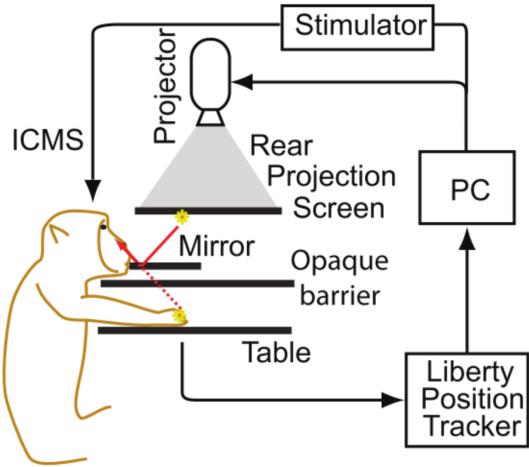


Figure 9 – Scheme of the set up of the experiment [1]

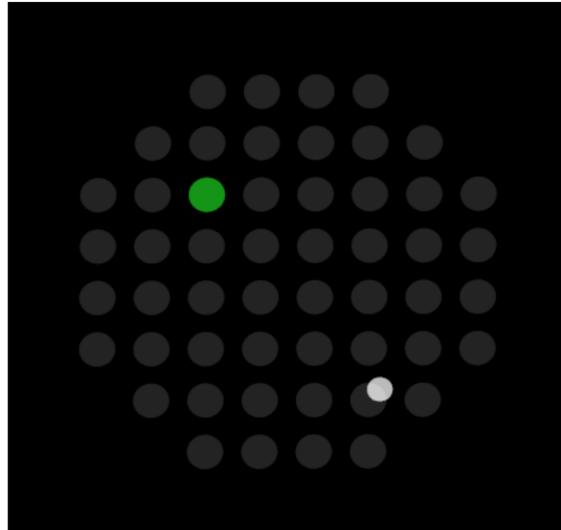


Figure 10 – Screenshot of the reaching task. The white dot represents the cursor position (hand position of the monkey located with the electromagnetic tracker), the green dot represents the target and the dark grey dots represent the possible locations for the target.

4.5 Artifact Removal

Electrical stimulation of neural tissue enables the implementation of brain-computer interfaces augmented with somatosensory feedback. However, a major challenge to address is the interference caused by undesired voltage transients resulting from electrical stimulation on concurrent electrical recordings. This undesired voltage transients, also called electrical stimulation artifacts, can be several orders of magnitude larger than the size of the physiological signals and can persist to hundreds of milliseconds after the termination of the stimulation. They represent a major issue as they disturb the online spike-sorting algorithm, which is essential for brain control. They lead to misinterpreted neural activity, distorted shape of extracellular potentials, disturbed detection and classification of action potentials. O'doherty, J.E developed a new technique to decrease the influence the Stimulation Artifacts on bidirectional neural interfaces during online processing [16].

Therefore, Stimulation Artifacts extraction is performed before running the BMI task. The stimulation pulse scheduler is monitored through a Matlab program, which goes through the sixteen stimulation channels, one at a time, sequentially, for four repetitions, and feed the artifact subtraction filter algorithm with the recorded artifacts.

4.6 BMI task

This task aims at demonstrating that the monkey is able to learn the meaning of a multichannel intracortical microstimulation signal (ICMS), and combine this artificial sensory feedback with visual feedback in order to improve his performance at the reaching task. This task is called BMI, which stands for Brain-Machine Interface, since the monkey is controlling a neural cursor on the screen and receives feedback from artificial neural stimulation.

BMI task implementation:

The BMI algorithm has been trained with the hand reaches task, and is able to determine the monkey's intention of movement from the neural activity recorded from the electrode array implanted in the motor cortex (M1), in order to move the neural cursor. Stimulation is provided through the electrode array implanted in the somatosensory cortex (S1). The stimulation delivers a continuous feedback about the position and the speed of the neural cursor on the screen. To avoid stimulation artifacts in the recordings, the results of the artifact removal step are taken into account in the algorithm.

In this BMI task, the monkey controls a neural cursor. His goal is to reach the green target that appears on the screen. If the left hand moves, the trial is automatically canceled. The monkey first learns to perform the task with VIS (visual feedback) alone. The task is similar to the hand reaches one. With time, we increase the transparency of the white dot (the neural cursor position) and increase the visibility of the lines pointing to the cursor's position. At the beginning, 100% of the lines points toward the neural cursor's position. Then, we add stimulation to the trials (VIS+ICMS). Spatiotemporal correlation between vision and ICMS leads to the integration of the artificial sensory feedback. The % of lines pointing to the neural cursor position decreases gradually over the training sessions, with the other lines pointing in random directions. Reducing the reliability of the visual feedback increases the relative value of the ICMS feedback. Over the course of several training sessions, the monkey is able to perform the task with 15% coherence (which means 15% of the lines pointing to the cursor's direction). (Fig. 11)

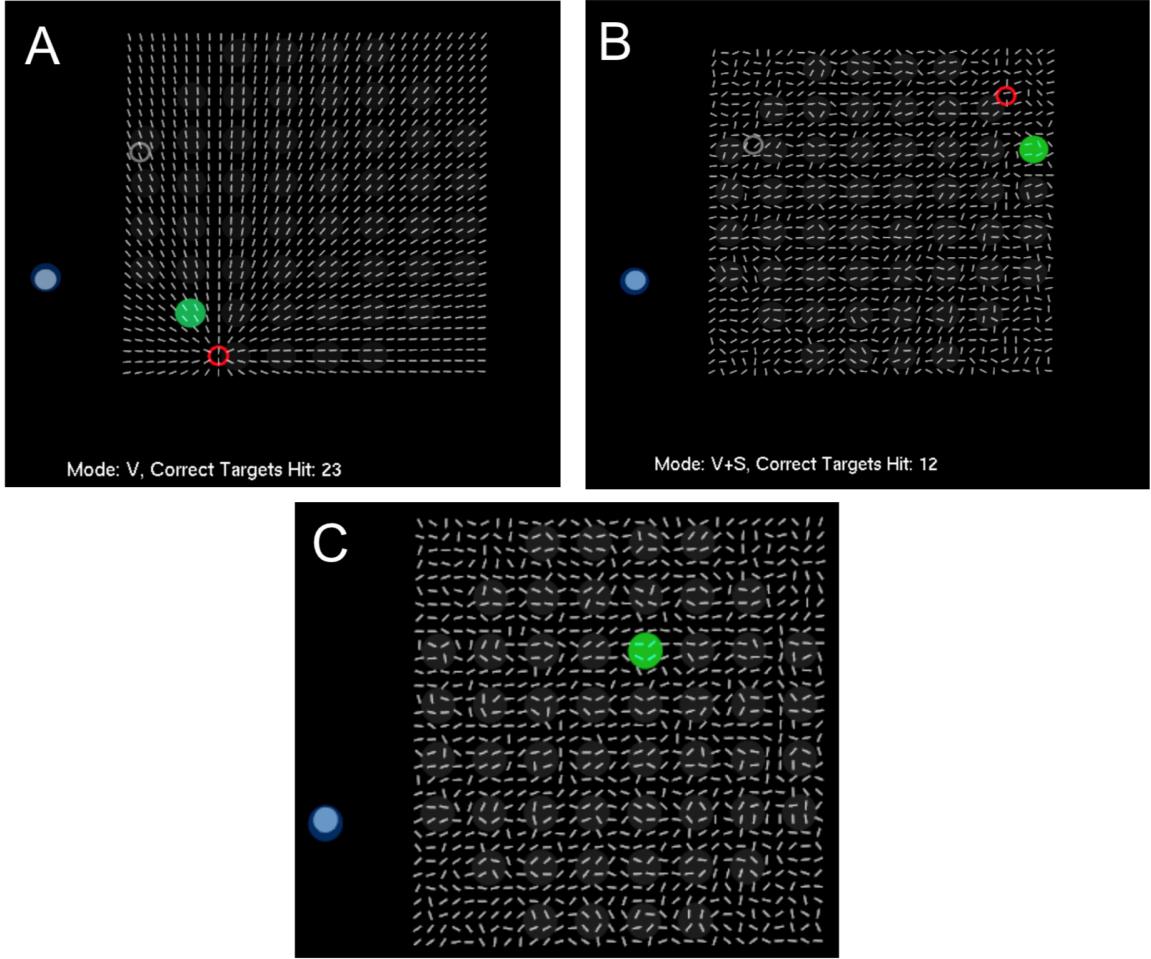


Figure 11 – A and B: Screenshots of the human operator computer monitoring the task. A: BMI task with 100% coherence. B: BMI task with 15% coherence. The green point is the target, the blue point represent the location of the left hand. Trial is canceled if the left hand moves. The red point is displayed for the human operator only, the monkey does not see it. It represents the position of the neural cursor controlled by the monkey. C: BMI task at 15% coherence. Corresponds to what the monkey actually sees.

4.7 Analysis of the performance at the BMI task

After a training session, we look at the performance of the monkey at the BMI task. To measure it, and assess the monkey’s progresses, the lab uses a set of different metrics. Among them, the time to complete the trial and the distance covered by the neural cursor are analyzed. Since the time to complete the trial largely depends on the distance of the target with the initial position of the neural cursor, we look at the normalized movement time and normalized path length.

Other parameters we can look at are the direction estimation, which is measured by regressing the movement angle when the monkey initiates the trial, against the real angle. Other interesting parameters would be distance estimation, mean and variance of the initial angle and the percentage of completed trials over the total number of trials.

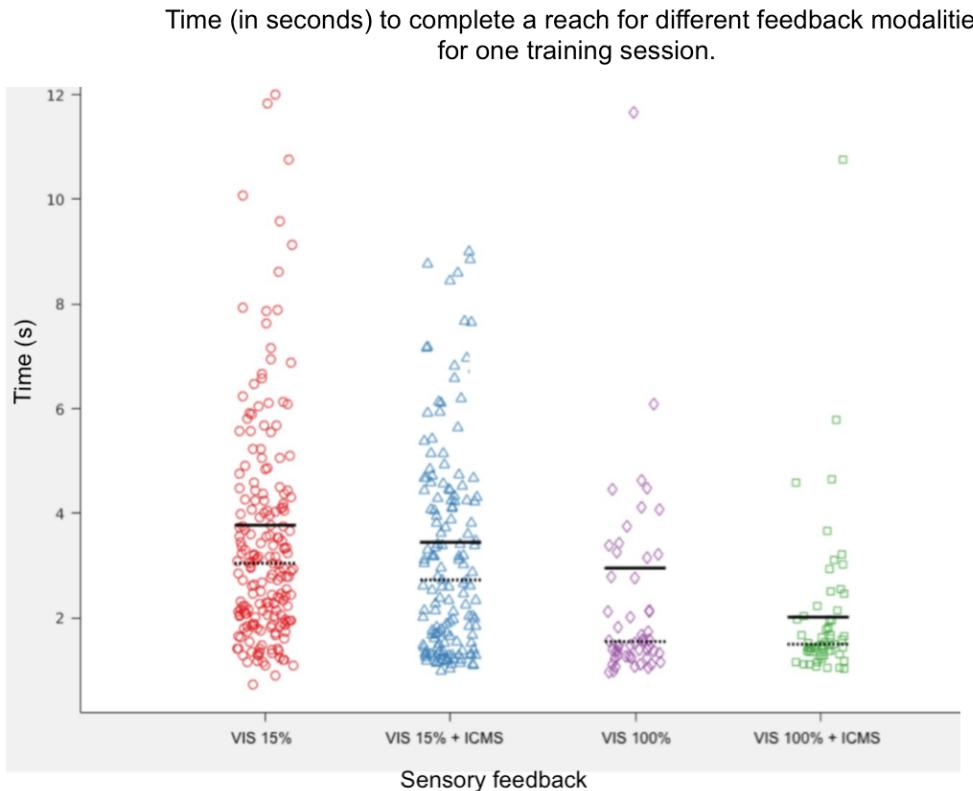


Figure 12 – Time (in seconds) to complete a reach for different feedback modalities for one training session.

Each point represents a trial. VIS 15% corresponds to trials with visual feedback alone and 15% coherence, same with VIS 100% with 100% coherence. VIS+ICMS stands for trials with visual and ICMS feedback. Black lines represent the median, and dot lines represent the mean, for every kind of modality.

This task aims at showing that the monkey improves his performance and optimizes his movement precision by combining information from the two sensory cues. VIS+ICMS trials become progressively better than VIS trials. On figure 12, we observe that VIS +ICMS trials are better than VIS trials only (the median and the mean of these modalities are lower). However, when I left the lab results were not significant yet and the monkey was still learning to integrate the ICMS feedback. Once that this study will be completed, it will show that ICMS feedback can be used as artificial proprioception (as it delivers continuous feedback about the position and speed of the neural cursor). Combined with brain control of the cursor (BMI), this technology could constitute an interesting strategy to restore proprioception in patients using BMIs in artificial limbs, exoskeletons, wheel chairs...

4.8 Detection task

The Detection Task that aims at assessing the sensitivity of the brain to the stimulated currents. Indeed, during the BMI task, some currents are delivered (intra-cortical micro- stimulation ICMS) through the electrodes located in the sensory cortex. The experiment consists into a two-target forced choice protocol. The monkey moves his hand onto a target to initiate the trial, and then moves to the left or right target depending if he felt a stimulus or not. More details are given in the Chapter Evolution of stimulation sensitivity across time.

5 Evolution of stimulation sensitivity across time

5.1 Introduction

Regularly (almost every week), we perform the Detection Task that aims at assessing the sensitivity of the brain to the stimulated currents. Indeed, during the brain control (BMI) task, some currents are delivered (intra-cortical micro-stimulation ICMS) through the electrodes located in the sensory cortex. We need to assess for each of the 16 electrodes what is the current at which 75% of the stimulus is detected and update the ICMS current amplitudes for the BMI task. My study focuses on the evolution of the sensitivity threshold of the 16 electrode pairs over time.

5.2 Methods

5.2.1 Monkey's task

The experiment consists into a two-target forced choice protocol. The monkey moves his hand onto a target to initiate the trial, and then moves to the left or right target depending if he felt a stimulus or not. For safety reasons, the current doesn't go over $100 \mu\text{A}$ in the detection task, and $75 \mu\text{A}$ in the BMI task.

5.2.2 Quest algorithm

The Detection Task is based on the Quest algorithm [17] to determine what is the next intensity of the current tested. We want to detect the current amplitude where the monkey can detect the stimulus with 75% chance. To detect this threshold, we need to test different current intensities. However, we have a limited amount of trials, so the Quest algorithm helps optimizing the choice of the next tested value in order to get the best estimate of the threshold T . Indeed, this method is more efficient than testing T on even spaced range scale, as for a limited amount of trial we spend more time around the critical point and this leads to a better estimate.

The Quest algorithm is a bayesian adaptative psychometric method. In other words, the algorithm uses the Bayes theorem, which describes the probability of an event based on prior knowledge of conditions that might be related to the event. For every new trial realized, the current most probable estimate of the threshold is updated. A psychometric function describes the relation between some physical measure (here the intensity of the current) of a stimulus and the probability of a particular psychophysical response (75% chance detection).

We have to take into account the sensitivity of neuronal cells. For the subject safety, we set the maximum current amplitude at 100 A . Other parameters are fixed before starting the Quest algorithm: the prior guess for the threshold value (set at $50 \mu\text{A}$ for the first session, and then set at the threshold determined at the previous session) and the steepness of the psychometric curve (set at 3.5, based on previous observations). The probability of a failure well above threshold is vanishingly small, but observers occasionally make errors regardless of how intense the stimulus. The possibility of these mistakes is taken into account with the parameter set to the value 0.01. The parameter specifies the probability of a success at zero intensity: for two-alternative forced choice it is 0.5.

5.2.3 Results

I collected the results of the Detection task from the 10/10/2016 to the 27/04/2017 (15 sessions). Results are plotted in Fig. 13. The position of the channels (= electrode pairs) on the electrode array is represented in Fig. 4.

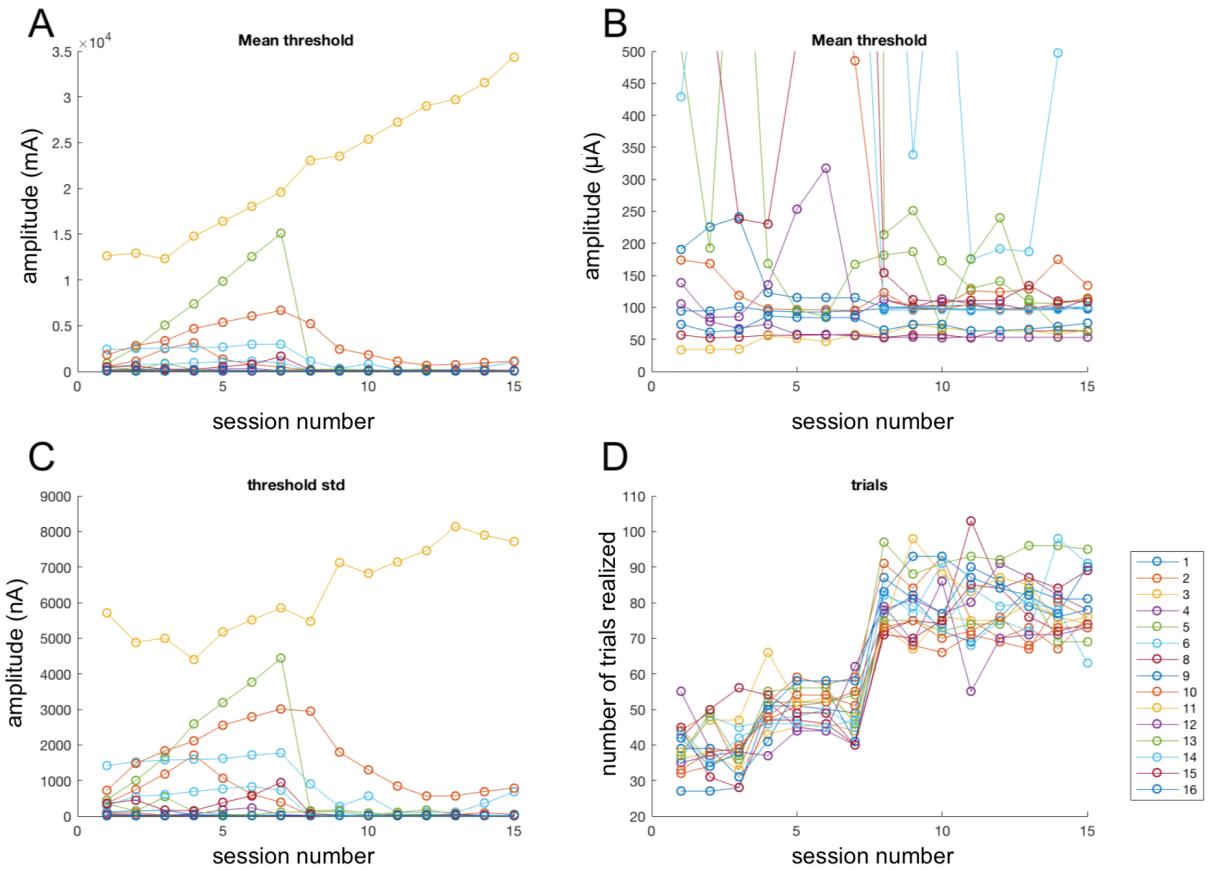


Figure 13 – Results of the Detection task from the 10/10/2016 to the 27/04/2017 (15 sessions). Every line corresponds to a different channel and each circle corresponds to a different session.
 A: Evolution of the threshold value across 15 sessions for 16 channels. B: zoom of graph A
 C: Evolution of the standard deviation of the threshold. D: number of trials realized.

We notice that the threshold point of 75% chance detection of the stimulus globally decreases across time (Fig.13A). One exception is made for channel 3 that keep increasing. At the 13th session, this electrode pair was replaced by an adjacent one, but no difference was noticed regarding the threshold. We observe that the threshold is mostly located under 150 μA (Fig.13B). Ideally, we would like to have all thresholds under 100 μA , because it is the highest stimulation current used, for safety reasons. On Fig.13C, we also observe that the standard deviation generally decreases (except for channel 3), which means that there is less variability in the results and the threshold is more accurate and stable. On Fig.13D, the number of trials per session is represented. With time, the monkey is able to be concentrated a longer amount of time and completes a higher number of trials.

6 Implementation of a Spike Sorting Algorithm

6.1 Neural Signals and Electrophysiological Recordings

A neuron is a cell that receives, processes, and transmits information through electrical and chemical signals. These signal transmissions between neurons occur via specialized connections called synapses. Neurons connect to each other to form neural networks, and they are major components of the central nervous system (brain and spinal cord) and the peripheral nervous system. Neurons are composed of dendrites that receive input signals, which are processed by the soma and information is transmitted through the axon in the form of action potentials (electrical impulses, also referred to as spikes). Recordings of the electrical signals emitted by neurons is referred to as electrophysiological recordings.

Electrophysiological recordings (intra and extracellular) are widely used in neuroscience. Studying the collective dynamics of multiple single neurons constituting large neural ensembles is fundamentals in many aspects of research related to sensory, motor and cognitive functions and can be used for rehabilitation purposes. This field of research is referred to as brain-control interfaces (BCI), brain-machines interfaces (BMI) or neuroprosthetics.

In the electrophysiological recordings, each electrode records the signals emitted by a group of neurons. Spike sorting is the process of separating this signal into groups of single-unit activity corresponding to single neurons [18]. It is a crucial step of neural data processing, and essential to train and control the brain control algorithm. In our experiment, electrodes implanted into the motor and sensory cortices record the extracellular tissue of the brain.

In order to separate the electrophysiological signal into single neurons activity, spike sorting relies on neuron's individual action potential shape. The variability in spike shapes is due to the type of neuron, and its distance and relative position to the recording electrode.

6.2 Requirements for the design of the algorithm

Spike sorting in the lab is performed by a human operator. Each spike is represented by a dot in a 2 dimensional space which axes are its two first principal components (explained in the Feature Selection and Dimensionality Reduction chapter). The operator sorts the neurons by drawing cluster boundaries with the help of the user interface (Figure 7).

Spike sorting technique in the lab should be standardized: manual control brings a lot of variability in labeling across different sorting sessions and different human operators. Moreover, the process of performing manual spike sorting every day before each experiment is time consuming. Furthermore, an automatic spike sorting algorithm can take into account more information from the signal and increase the sorting accuracy. Also, the human spike sorter will never keep up with the increasing volume of data arising from increasingly large electrode arrays applied over increasingly long durations. As the processing of the amount of data generated becomes a limiting factor, it is urgent to implement robust methods to process data online without any user intervention. These observations led me to the goal of implementing an automated spike-sorting algorithm.

While designing the algorithm, I had to take into account many requirements. First, the algorithm should be able to run on real time for online brain-control. The designed algorithm includes an offline training period before the real-time classification period. Second, the algorithm should be insensitive to differences in dataset properties such as the cluster's shapes, the firing rates, etc. The model should be unsupervised, and minimizing both the number of user-defined parameters and the number of modeling assumptions, to fit to any structure of the data, while maintaining high spike sorting accuracy and efficiency. Ideally, this fully-automatic algorithm should also be

computationally efficient, with the perspective of having this processing step fully embedded into the recording devices themselves in the future. For implantable devices, it is particularly important to not heat the surrounding biological tissue and optimize the bandwidth for wireless systems.

6.3 Challenges in Spike sorting

Classifying neural data is particularly difficult, and the main reason is the lack of ground truth. It is not possible to use known data to assess the sorting performance of the classifier. To remedy to this issue, one solution would be to generate an artificial dataset [23-24]. However, there is no simple noise model: the background signal arises from the combinations of multiple complex signals, including small spikes from hundreds of distant neurons, and can contain electrical noise mixed with true neural signals. Some studies also use post-processing techniques based on the knowledge of the statistical properties of firing neurons. For instance, they analyze the distribution of interspike intervals (ISIs) the, standard deviation (S.D.) test, the χ^2 test, or the projection test [25]. Other challenges need to be taken into account while performing spike sorting, as time- overlapping spike signals, or variation in the spike shape, for instance when the cell position drift over time relative to the physical electrode, or when bursting occurs[26-27].

For this project, I focused on clustering algorithms such as k-means, expectation-maximization on mixture of Gaussian, the Isosplit method applied to the Gaussian Mixture Model, and the Density-Based Spatial Clustering algorithm.

All the analysis has been performed on the spike file from the Quiet Sitting task of May 5th, 2017.

6.4 Pre-processing of the data

First, the data is filtered with a linear phase high pass FIR filter (300Hz) to remove the DC component without phase distortion. Then, an IIR low pass filter is applied (5kHz) to remove high frequency noise. Spikes are detected when the filtered neural signal crosses the threshold value, which is 3.95 times the standard deviation. The detected events are aligned with their minima (temporal alignment is essential to compare the spikes). The extracted waveforms have 64 points sampled at 24.4kHz, which means they last $2.6 \mu\text{s}$. I preserved this part of the lab's algorithm and focused on the features selection, dimensionality reduction and clustering steps.

6.5 Feature Selection and Dimensionality Reduction

A feature is an individual measurable property of the process being observed. The bigger the number of features, the more complicated it is to extract significant parameters to perform an efficient analysis of the data. The need for efficient feature selection algorithms has emerged for several reasons. First, it aims at reducing the size and complexity of the problem, and consequently reducing the computing time and the required memory to run our algorithms. Furthermore, when a clustering algorithm follows feature selection, the accuracy of the classifier is improved by removing irrelevant features that would increase the likelihood of overfitting to noisy data.

An efficient feature selection method should be able to select the features that contain the maximum discrimination to separate the data points into clusters, while avoiding redundant information brought by highly correlated features. The advantage of feature selection is that it does not alter the original representation of the features, but merely select a subset of them, offering the advantage of interpretability, but is at the cost of performance.

An alternative to feature selection is feature extraction. This process consists on creating new features from the original ones, with the extraction of instructive and non-redundant information. The study of dimensionality reduction method is restricted here to techniques that do not need to preserve the topological properties of the input space, and can be applied to unsupervised approaches. Among these techniques there are the Principal Component Analysis (PCA), Kernel PCA.

The PCA process involves projecting the high dimensional sample onto a space of lower dimensionality that carries sufficient information. The resulting features are a combination of the former features. Dimensionality reduction of the input space might result in a loss of information for the discrimination of the classes. The challenge relies on preserving as much relevant information as possible.

Vapnik-Chervonenkis theory maintains that mappings that take us into a higher dimensional space than the dimension of the input space provide us with greater classification power. The Kernel trick enables to benefit from high dimension whilst avoiding high computation. Therefore, Kernel PCA is a PCA applied to a high dimensional feature space using the kernel trick. A Kernel function projects the data points in some nonlinear feature space of higher dimension. Points can be separated in a non-linear way and different kernel function can be used as the Gaussian, Hyperbolic and Tangent ones. Kernel PCA can give a good re-encoding of the data when it lies along a non-linear manifold. However, the kernel matrix is NxN, so kernel PCA will have difficulties if we have lots of data points. Therefore, I selected PCA for the feature extraction process.

After some research, I realized that most of the spikes sorting algorithms rely on PCA for feature selection and dimensionality reduction as well [17]. Principal Components Analysis is a statistical procedure that uses an orthogonal transformation to transform a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables. PCA captures the directions in the data with the largest variation, through eigenvalue decomposition of the covariance matrix of the data. Principal components are sorted by the amount of variance captured. Therefore, the dimensionality of the dataset can be through a selection of the first principal components.

The extracted waveforms have 64 points sampled at 24.4kHz. These points are referred to as the features of the spikes. PCA features do not correspond to any physical feature and direct interpretation is complicated. This is why I tried to introduce real physical features, even if it is considered as an obsolete method. Among the 23 physical features considered, there are the first 10 coefficients of the discrete Fourier transform, the variance and mean of the signal, the maximum and minimum value their indices, the width between these two extrema, the peak to peak amplitude, and other parameters describing the derivative of the signal.

Combining these 23 physical features with PCA improved the separation of clusters, in comparison than projecting directly the raw data on the main principal components (Fig. 14). After defining the new feature space, the number of components has to be determined.

The dimensionality is restricted to the first 3 principal components (PCs), after applying the Gaussian Mixture Model (explain in the next chapter) for clustering the spikes. We noticed that the difference between the spike shapes of the clusters is more accentuated when taking into account only 3 Principal Components. At some point, adding components increase the noise of the signal and decrease the quality of the spike sorting (Fig. 15).

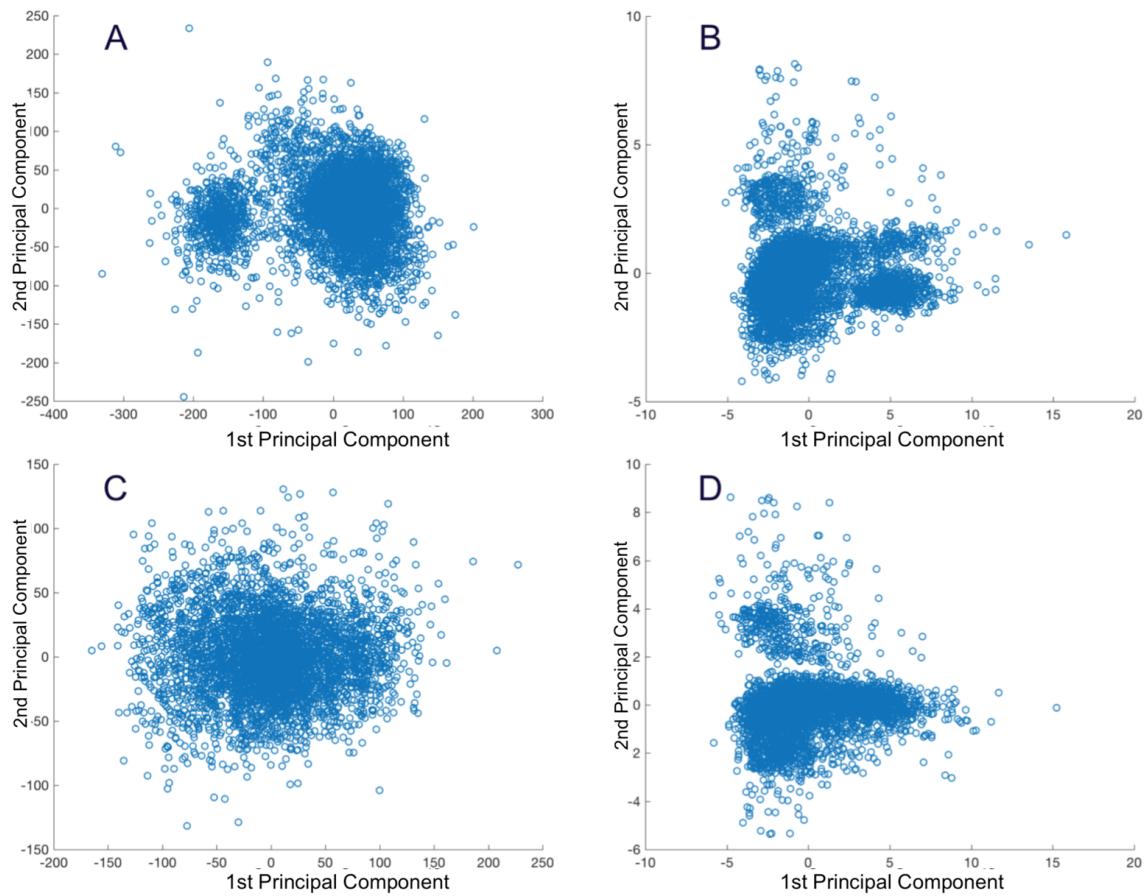


Figure 14 – Comparison of 2 different methods of dimensionality reduction: PCA from raw data (A,C), PCA from 23 physical features (B,D). Channel 6 (A,B) and 74 (C,D). The data is projected on the 2 first Principal Components for visualization, however the algorithm takes into account more components.

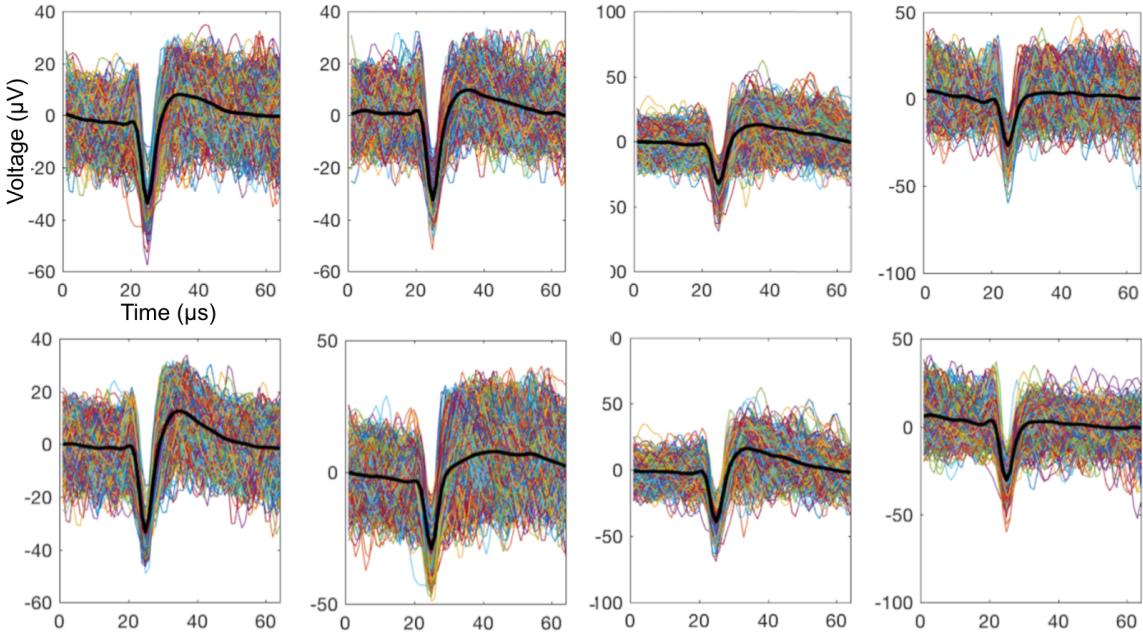


Figure 15 – Spikes of the clusters obtained with the Gaussian Mixture Model algorithm on Channel 4, with 4 clusters. On the first line, 10 principal components were taken into account. On the second one, only the first 3 principal components were considered. The black waveform represents the mean of all the spikes identified on this clusters.

6.6 Clustering

Now that we selected the relevant features and reduced the dimensionality of our dataset, we want to assign the spikes to different neurons. Unsupervised clustering is definitively the most complex part of the algorithm. A wide variety of methods are available, some of them making assumptions on the data, and choices need to be made regarding the tradeoff between accuracy and complexity.

6.6.1 Gaussian Mixture Model

The Gaussian Mixture Model (GMM) is a clustering algorithm, which uses an iterative technique called Expectation-Maximization, where data points are assigned to the clusters regarding their probability of belonging to each distribution. All clusters are assumed to have an independent Gaussian distribution and their own mean and covariance matrix. This method is preferred over k-mean clustering which relies on straight Euclidean distance, because the clusters are assumed to be spherical with similar covariance.

The GMM algorithm is initiated with random data points as initial means, covariance matrices for each cluster equal to the covariance of the full training set, and equal prior probability.

Then, the GMM algorithm uses an iterative technique composed of 2 steps. In the Expectation step, the probability of each data point to belong to each cluster is computed, using the current estimates of mean vectors and covariance matrices. In the Maximization step, the cluster means and covariances are re-calculated, based on the probabilities calculated from the Expectation step.

The main issue with the Gaussian Mixture Model algorithm is that the number of clusters has to be tuned beforehand. One way to solve the problem of the number of clusters is to try all the

models with a number of clusters varying from 1 to 4, and then compare their relevance. The first idea would be analyzing the likelihood, but this measure would always select the highest number of clusters. Another solution would be using the Akaike Information Criterion (AIC). AIC provides a method for model selection, and introduce a trade-off between the goodness of fit (represented by the log-likelihood $\ln(L)$) and the complexity and overfitting of the model (represented by the number of parameters k). The Bayesian information criterion (BIC) was also tested. This method uses the number of parameters k and also the number of observations n as a balance over the goodness of fit. The smaller these criteria are, the better is the model.

$$AIC = 2k - 2\ln(L) \quad (1)$$

$$BIC = \ln(n)k - 2\ln(L) \quad (2)$$

However, these models were not working: On 10 channels, with a cluster range from 1 to 4, the model with the highest number of cluster would always be selected (Fig 16). One option would be changing the coefficient multiplying the number of parameters in the AIC model (Fig. 17), but I haven't found a satisfying and automatic way of selecting it. Also, computing 4 times the model with different number of clusters in order to choose the best model requires a lot of computing resources, and I decided to find a more optimal way to sort the spikes into different neurons.

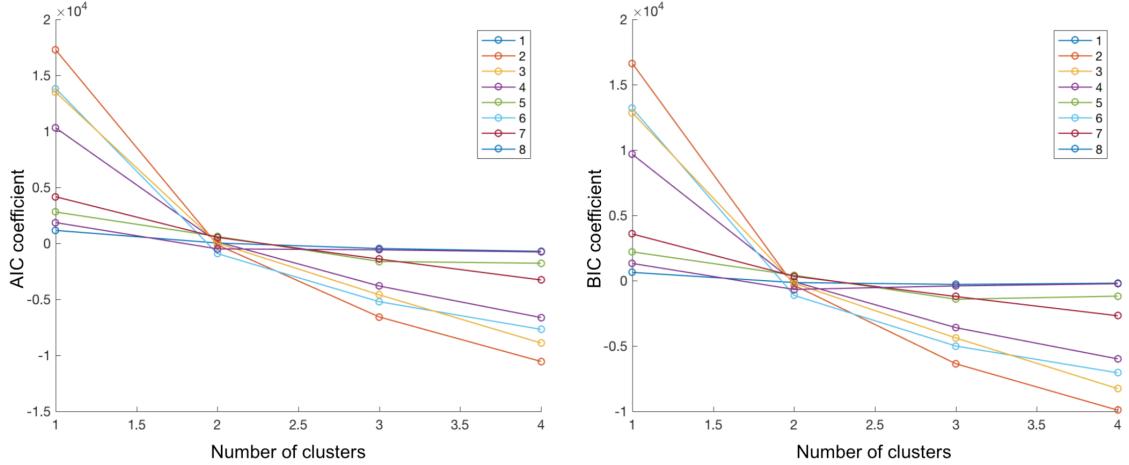


Figure 16 – Akaike Information Criterion (left) and Bayesian information criterion (right) for 8 channels. Each line corresponds to a different channel. The model with the highest number of clusters is always preferred.

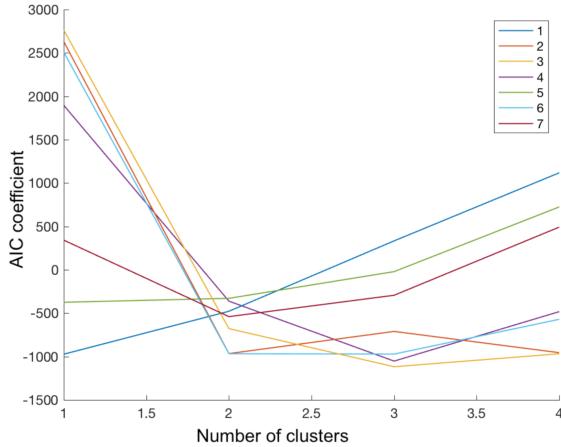


Figure 17 – AIC with a coefficient of 1000 for 7 channels . Different numbers of clusters are selected for each channel, however it is difficult to tune the coefficient in a non arbitrary way.

6.6.2 Gaussian Mixture Model with Isosplit

Since the first approach was not successful, and a simple GMM algorithm needs a priori information about the expected number of clusters, I tried another idea inspired from Magland J.F., Jason E Chung J.E. & al [19]. Their approach called Isosplit relies on very fine parcellation, which substantially overclusters the data. They assumed that each cluster arises from a density function which, when projected onto any line, is unimodal, having a single region of highest density. Furthermore, two distinct clusters may be separated by a hyperplane, in the neighborhood of which there is a relatively lower density. Therefore, the next step of the algorithm is to iteratively compare the clusters by pairs. Their points are projected onto a line connecting their centroids, and a statistical test (Hartigan dip test [20]) is performed to test whether the distribution is unimodal or multimodal. If this statistical test determines that the null hypothesis of unimodality is accepted, the clusters are merged. The iterations continue until all pairs of clusters have been tested.

The Hartigan's dip test calculate the empirical distribution function EDF and the cumulative distribution function CDF of the data based on the population mean and variance, the greatest convex minorant (GCM) and least concave majorant (LCM). The CDF of a unimodal distribution will only have one mode: it is convex before the mode, and concave after. The CDF of a multimodal distribution will have more than one mode, and therefore have regions alternating between concave and convex. The Hartigan's dip test performs an analysis of the maximum distance (dip) between the EDF and CDF.

In figure 18, the computation of the Hartigan's dip statistic from a range of Gaussian distributions has been realized, to get an insight of the dip and p values for unimodal and bimodal distributions.

The issue with this method is the variability of the final cluster distribution (Fig. 19). Maybe a higher number of initial clusters and a better tuning of the parameters would solve it. The tuning conditions (dip and p-value) to merge the clusters were hard to define with real data, and generalization for all channels in order to have an efficient sorting was tricky too. Furthermore, with a high number of initial clusters (15-20 clusters) the covariance can be ill conditioned, and the program returns an error. So the initial number of clusters was limited to 10.

Even though I noticed an important variability of the final cluster distribution with this method, I compared it to a simple GMM algorithm, and assessed the efficiency of the isosplit (Fig. 20). The difference between spikes waveforms seems higher in the GMM algorithm, and the isosplit requires extra computational power.

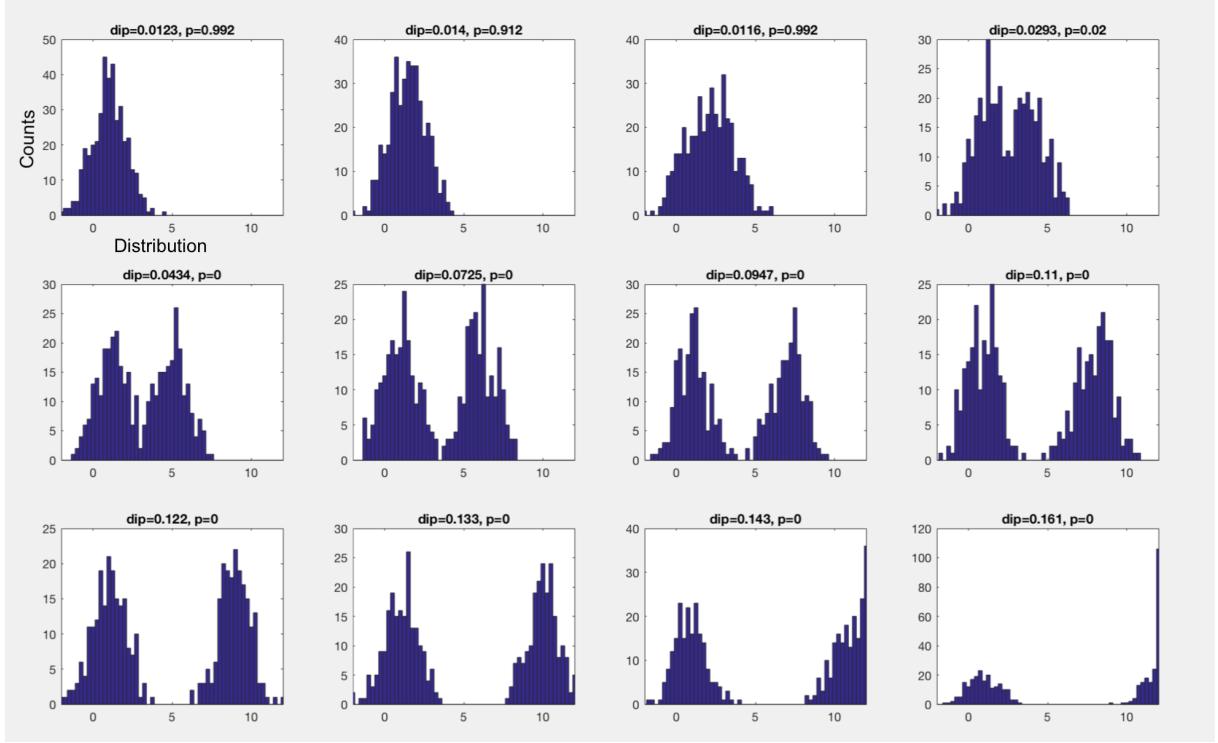


Figure 18 – Computation of the Hartigan’s DIP statistic from a range of Gaussian distributions, to get an insight of the dip and p values for unimodal and bimodal distributions. For each case, the dip and the p value are calculated.

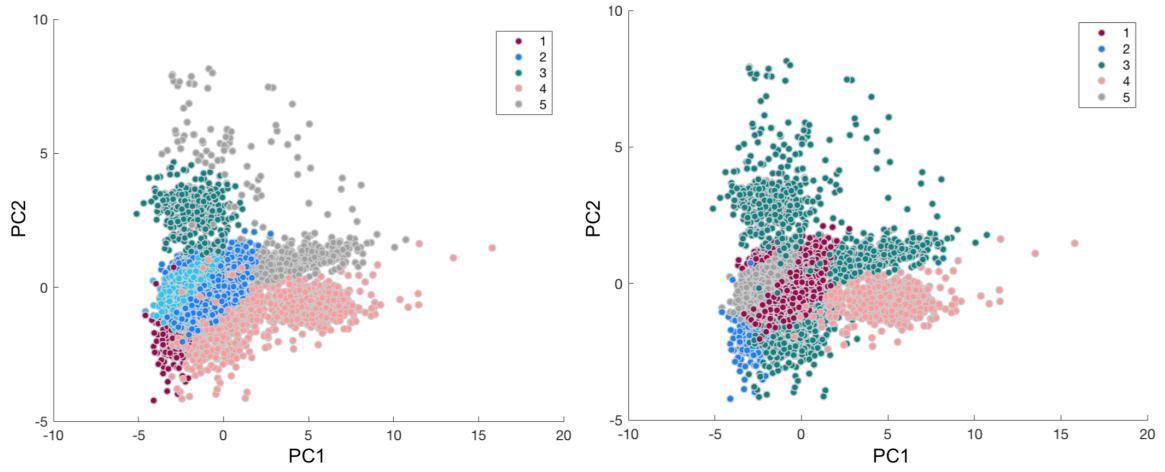


Figure 19 – Important variability across the clusters through the GMM and Isosplit method. Results show the clusters formed by the data of channel 4, with an initialization at 10 clusters. Data is projected into the 2 first principal components. Colors correspond to the different clusters.

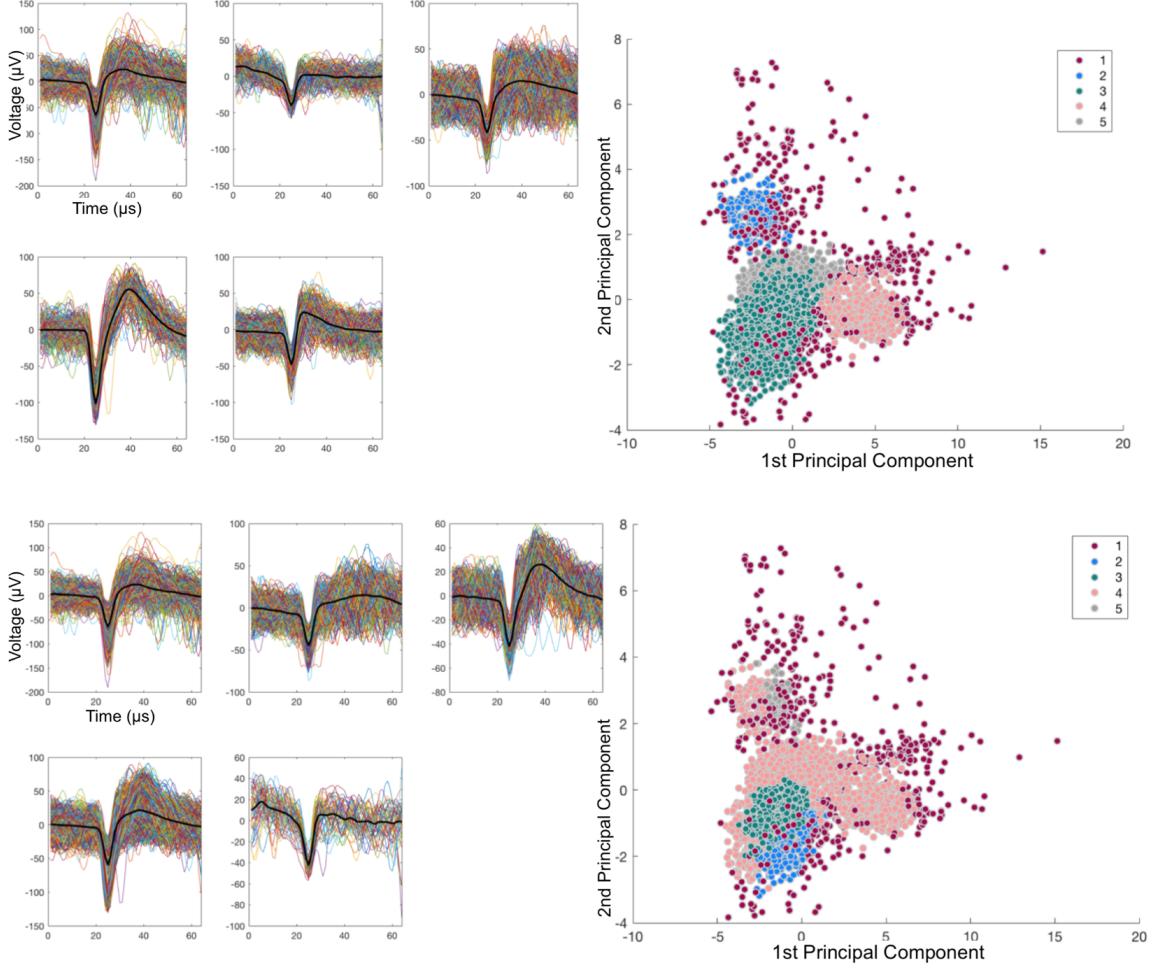


Figure 20 – Top: GMM with 4 clusters. Bottom: GMM with 10 clusters and Isosplit. On the left, spikes identified for each cluster. The black waveform represents the mean of all the spikes identified on this cluster. On the right, data projected into the 2 first principal components. Colors correspond to different clusters. Cluster 1 is identified as noise.

6.6.3 Density-based spatial clustering algorithm

The density-based spatial clustering (DBSCAN) algorithm [21] appears to be the best candidate for the clustering step of the algorithm. Indeed, it doesn't require input parameters such as the number of clusters, and it is also able to determine clusters of different sizes and shapes, as well as recognizing noise and outliers. It is also a much faster process than the one considered previously, since it neither requires to run models over a set of parameters values to choose the best fitting ones, nor it operates with a recurrent pattern an important amount of times across all the data points. Furthermore, DBSCAN doesn't assume that clusters have a Gaussian distribution, and can handle clusters of different shapes and sizes.

Explanation of the algorithm

DBSCAN is a density based algorithm. It starts by selecting an arbitrary point within the data points, and finds all the neighbor points within a distance R of the selected point. This radius R represents the size of the neighborhood. If the number of neighbors is bigger than N , which is the minimum number of neighbors required to consider a data point as a core point from a cluster, the visited point and its neighbors are added to the cluster. However, if the number of neighbors is

smaller than N , the visited point is labeled as noise and the algorithm selects another data point that hasn't been visited yet. The algorithm repeats this evaluation process until all the points have been visited. Figure 21.

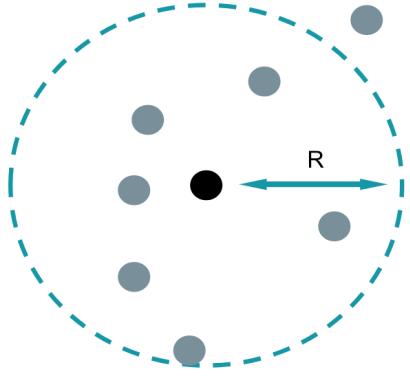


Figure 21 – Scheme of the DSCAN algorithm procedure. Considered data point is represented in black, and neighbor data points in grey. R is the neighborhood radius.

Tuning of the parameters

In order to use the DBSCAN algorithm without human supervision, we need an automated way to define the values of the radius R and the minimum neighbors N , in function of the dataset. To determine the radius R , we look at the behavior of the function representing the distance from the points to their k -th nearest neighbor, which is called k -dist. The function representing the distance of all data points to their 1st nearest neighbor is displayed in Figure 22. We notice that there is just one inflection, which means all the clusters have the same density (same observation is made for the other channels). The distance at which there is a sharp change in this graph (detected automatically through the analysis of the 2nd derivative) is equal to $R/2$. N is defined in function of the mean distance of the 1st nearest neighbor and the radius. Results obtained with this method are shown in Figure 23 and 24.

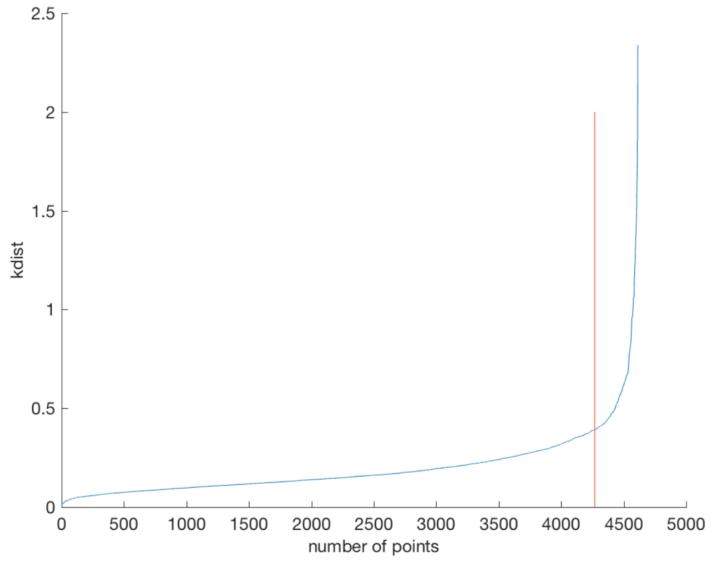


Figure 22 – DBSCAN algorithm: tuning the parameters. Function showing the distance of the data points with their 1st nearest neighbor, for one channel. The distance is represented on the vertical axis, and the horizontal axis represents the number of points that have their first neighbor within this distance. The red line represents the detection of the inflexion point in the curve. The distance at which the blue curve is intersected with the red line is $R/2$.

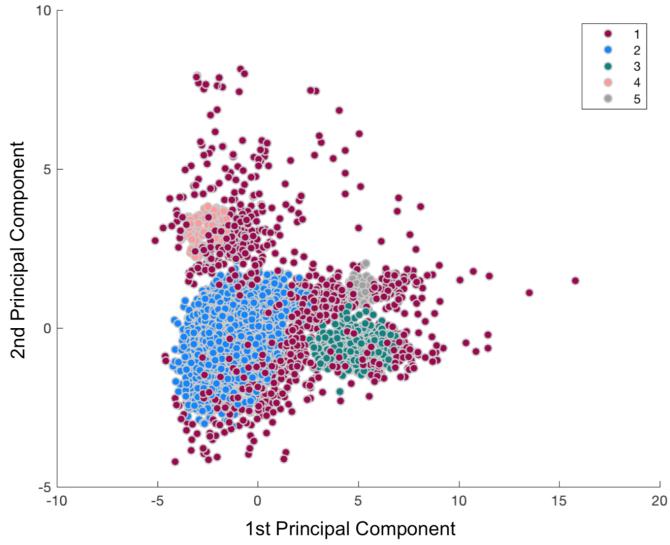


Figure 23 – DBSCAN algorithm: Clusters determined with the DBSCAN algorithm for one channel. The radius is 0.74 and N is 39. The data is being projected into the 2 first principal components. Colors correspond to different clusters.

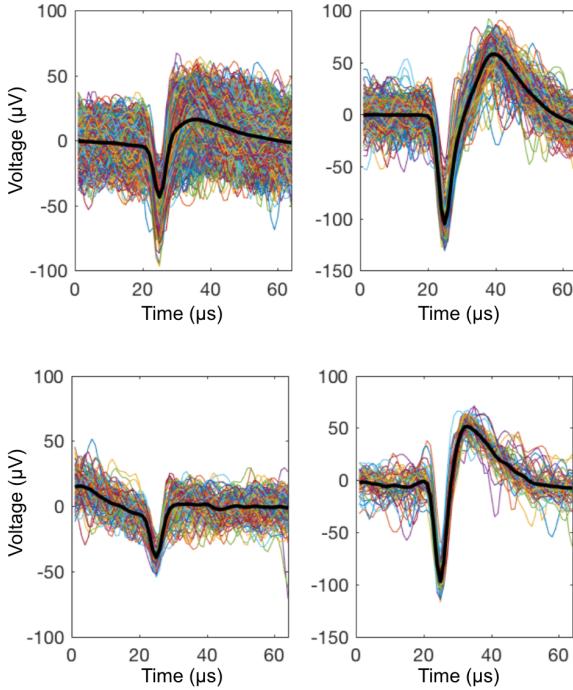


Figure 24 – DBSCAN algorithm: Spikes corresponding to the identified clusters of Figure 23 by the algorithm for one channel. The black waveform represents the mean of all the spikes identified on this cluster. The first spikes are the ones labeled as noise.

I concluded that the DBSCAN is the most appropriated algorithm for spike sorting. It doesn't make any assumption on the data and cluster shapes, it is able to detect and isolate noise, and it doesn't make any assumption on the number of clusters. Furthermore, the algorithm is relatively simple, it doesn't require much computational power and is really fast.

6.7 Further Perspectives in Spike sorting

Many other models exist for spike sorting algorithm, such as t-distributions, C-means, superparamagnetic clustering, mean shift, online template matching and deep learning models that I did not have the time to study. A very recent paper from M. Bernet and B. Yvert titled *Fully unsupervised online spike sorting based on an artificial spiking neural network* and published in 2017 [29] implemented successfully a novel spike sorting method based on the use of an artificial spiking neural network implementing different plasticity rules. By contrast with classical spike sorting approaches, this method does not use the conventional three-step procedure. Instead, the signal stream is continuously fed to the network, which rapidly learns to simultaneously detect and sort action potentials present in the signal thanks to its unsupervised learning properties. Once learning is stabilized, the output spike trains of the network predict those of the different active cells embedded in the input signal.

Classifying neural data is complicated because of the impossibility to know the ground truth in the cluster separation, however improvements in the generation of artificial dataset could help assessing the performance of the clustering algorithm.

It would also be interesting to consider in the model the firing rate of neurons and the time elapsed between the spikes (referred to as interspike intervals (ISIs)). Indeed, sometimes neurons emit action potentials in brief bursts of high frequency discharge, and these spike bursts constitute an important element in synaptic plasticity and information processing in the central nervous system.

Neurons generate bursts to increase the reliability of the communication and avoid synaptic transmission failure. Burst analysis might provide more precise information than single action potentials and convey selective communication between neurons based on resonance frequencies. We could therefore implement a model that takes into account parameters such as burst patterns, decreased burst frequency, firing activity, number of bursts and burst duration, the number of spikes in a burst, the percentage of spikes in a burst, the inter-spike and inter-burst intervals.

Electrodes record signal from a pool of neurons, and sometimes these neurons trigger action potentials more or less at the same time, and these signals overlap, therefore preventing the detection of the spikes triggered by these neurons. Developing a model able to detect overlapping signals and recreating the corresponding pattern seems very complex, however M. Bernet tried to overcome this problem by the implementation of plasticity rules in her model [29] .

Furthermore, over time we notice a variation of the spike shape, when the cell position drift over time relative to the physical electrode, or when bursting occurs and different plasticity rules modify the spike shapes [26-27]. At last, interferences caused by undesired voltage transients resulting from electrical stimulation on concurrent electrical recordings are recorded. These stimulation artifacts can be several orders of magnitude larger than the size of the physiological signals and can persist to hundreds of milliseconds after the termination of the stimulation. They represent a major issue as they disturb the online spike-sorting algorithm. They lead to misinterpreted neural activity, distorted shape of extracellular potentials, disturbed detection and classification of action potentials. An approach to try overcoming this problem has been developed by O'doherty [16].

7 Craniotomy surgery

The laboratory performed a craniotomy surgery on an adult male rhesus macaque monkey to implant two 96 channel-silicon microelectrode arrays (Blackrock Microsystems, Salt Lake City, UT) into the left sensorimotor cortex. One array was implanted over the primary somatosensory cortex (Brodmann area 1, S1) and the other one in the primary motor cortex (Brodmann area 4, M1). Areas with receptive fields spanning in the right arm and shoulder were targeted. In this chapter, I summarize the Craniotomy Surgery I had the opportunity to assist to. This surgery, which spanned over 10 hours, enabled me to learn about the craniotomy process and all the steps required to implant electrode arrays on the surface of the cortex. For a full understanding of the process, I completed my knowledge by reading the paper published by Oliveira, L. M., & Dimitrov, D., *Surgical techniques for chronic implantation of microwire arrays in rodents and primates*, which provides a detailed guide of the surgical protocol [28].

7.1 Preoperative animal preparation and Anesthesia techniques

The primate is fasted the day before the procedure, to lessen the chance of vomiting and aspirating into the lungs during administration of anesthesia and placement of the endotracheal tube. At the beginning of the procedure, the animal is anesthetized with a combination of Ketamine/Midazolam given intra-muscularly and followed by inhaled 2-3% Isoflurane through an endotracheal tube. Indeed, general anesthesia suppresses the central nervous system activity and results in unconsciousness and total lack of sensation. Then, the animal is catheterized for fluid delivery in either saphenous or cephalic veins, his hair is shaved on the head and he is prepared for surgery. Analgesics (Meloxicam and Buprenorphine) are given following sedation to provide multimodal pain relief. Intraoperative antibiotic (cephazolin) is given every 2 hours during procedure to prevent infection.

7.2 Intraoperative monitoring

During the surgery, CO₂ measurements (through pulse oximetry), blood pressure and the respiratory rate are monitored continuously as they correspond well with swelling and retraction of the brain; CO₂ being the main driver of vascular dilation and constriction. The respiratory rate is adjusted to keep CO₂ in the physiologic range. If brain swelling or retraction occurs, adjustments to the ventilation can be made and intravenous narcotics can be given to depress respiratory drive and counteract hyperventilation. The depth of anesthesia is also controlled through the heart rate. This parameter is carefully monitored with an electrocardiograph to know the adequate level of anesthesia and the heart rate above which the animal is likely to begin moving. Finally, core body temperature is as well monitored and kept in the physiologic range.

7.3 Implantation techniques

The monkey's head is placed in a stereotaxic frame to provide physical stability to perform the surgery. The hair on the head has been shaved. A skin incision is made midline with extensions laterally to enable adequate exposure. Other soft tissues down to the cortical bone (the muscle and the periosteum) are cleared off the skull bone. The area of skull that will be opened is selected and marked with a pen. The squared-shape area of the skull selected is then removed, with the help of a dental drill, and placed in a sterile saline solution. The dura is then incised and retracted to expose the underlying cortex and allow visualization of the sulcal landmarks that guide the arrays placement. The screw holes for the connectors are drilled (this step comes before the placement of the electrodes to avoid vibration of electrodes after they are implanted in the brain). The array connector is attached to the skull with titanium screws. At this point, the position of the arrays is defined very precisely and a machine performs a high-speed pneumatic insertion through the pia-arachnoid and into the cortex. The arrays are connected through gold wires to connectors

that will be in contact with the headstages. The gold wires are fixed with dog bones, and the connectors are screwed to the skull. Then, a piece of Teflon is inserted between the Dura matter and the brain, in order to avoid a scar reaction that might interfere with the signal. The dura matter is sewed; the bone flap is placed back and fixed with screws. Dental acrylic cement is used to cover the surgery area and prevent infections. Indeed, with a previous monkey brain surgery in which the skin was sewed, the monkey experienced an infection 4 months after the surgery. After the cement has completely hardened, the skin is placed tightly around the head cap.

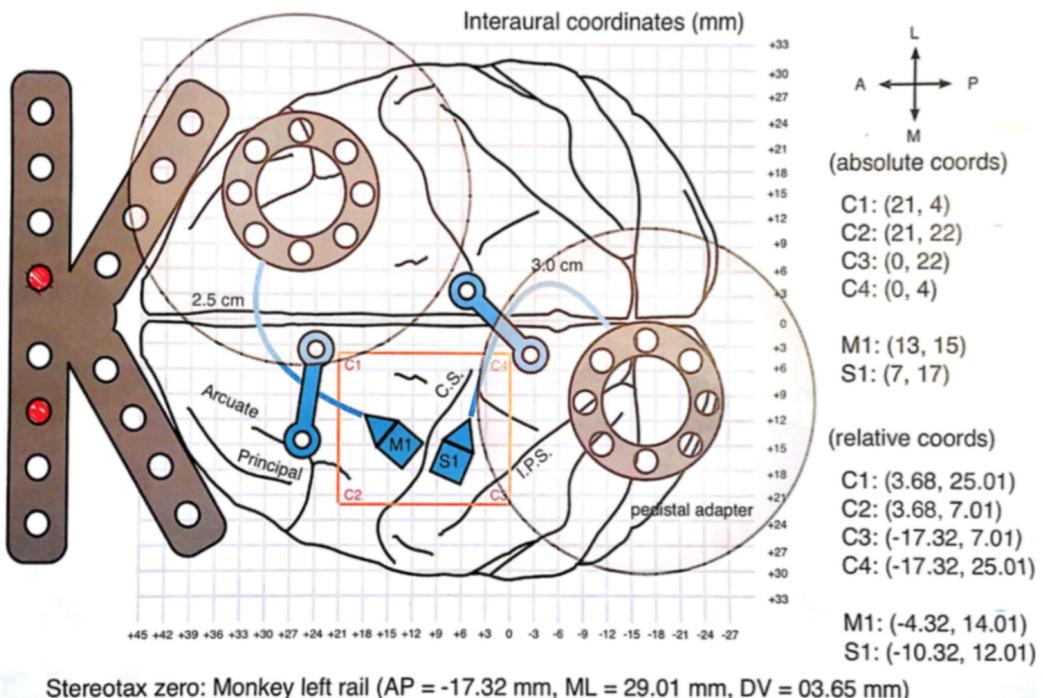


Figure 25 – Areas targeted for the electrode arrays (one in M1, the other in S1).

7.4 Postoperative care

Upon completion of surgery, the animal is recovered with O₂ and returned to vivarium cage with heat source. Primate is monitored continuously until moving/perching freely. Wet food and juice are left for recovering. Antibiotics and analgesics are given post operatively for 7-10 days (time to recover from surgery). Contribution to the surgery procedure

During the surgery, I was helping with the monitoring of the animal throughout, writing the values of a set of parameters (Blood Pressure, Body Temperature, Mean Arterial Pressure, Respiratory Rate, amount of dissolve CO₂ in expire breath (EtCO₂), peripheral oxygen saturation (SpO₂), % of inhaled isofluorane) every 15 minutes. Analysis of the electrodes receptive fields after implantation of the arrays

Once that the monkey has recovered from the craniotomy in which he had electrode arrays implanted on the left side of M1 and S1, we have to check whether the electrodes receptive fields are spanning into the right arm and hand. This step is essential for brain control: the movement of the arm is correlated to the neural activity of the motor cortex area responsible of this part of the body. Moreover, it is relevant to stimulate in the area encoding for the arm in the somatosensory cortex.

In order to do detect the receptive field of each electrode, for each electrode, we visualize its

broadband on the computer screen and plug the speakers to its signal. We touch, rub, and move the monkey's right arm, hand, shoulder, fingers, elbow and chest. While doing it, we listen and visualize when we obtain a bursting activity of the neurons recorded by the selected electrode. These results are reported on Figure 26.

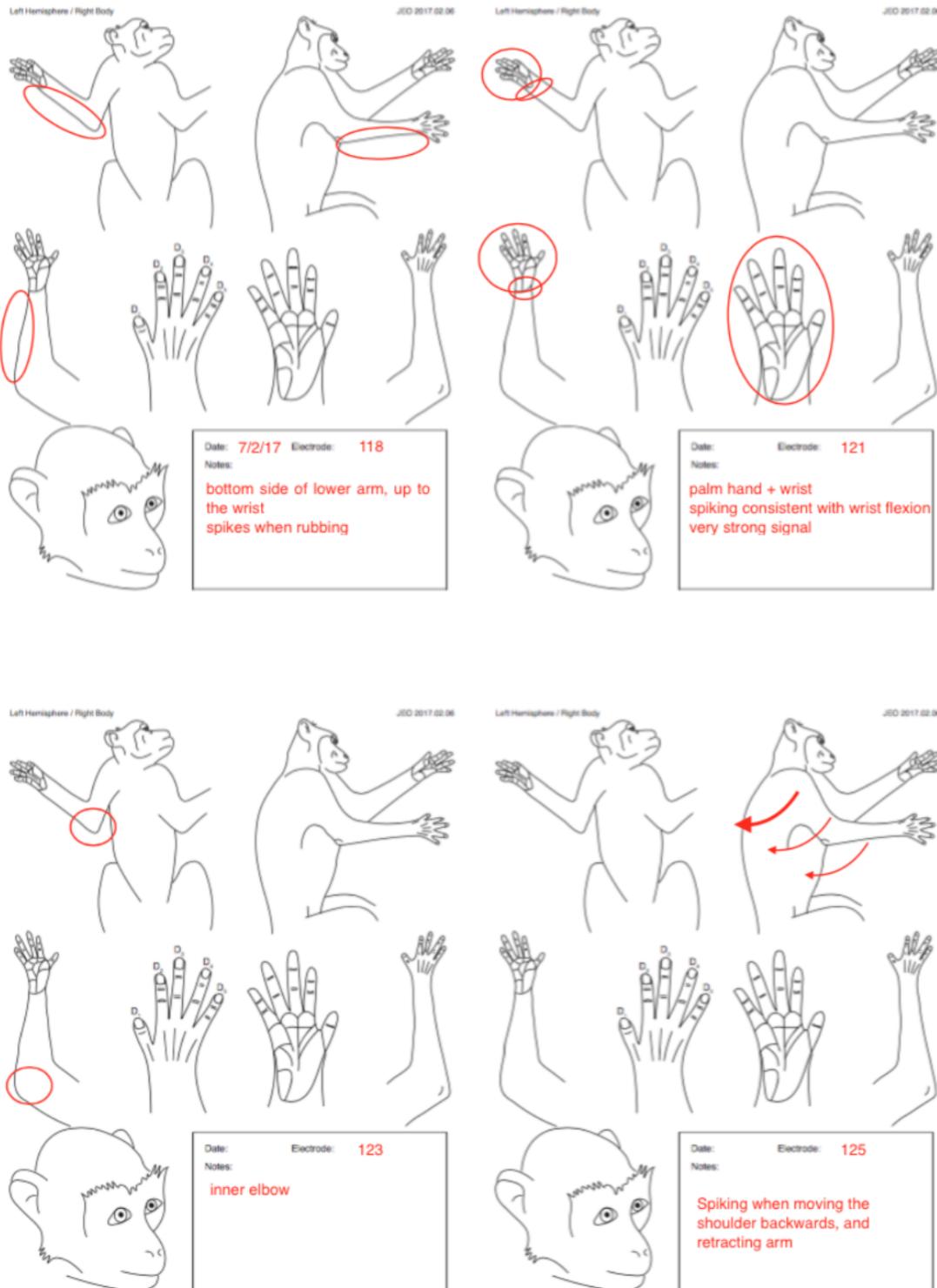


Figure 26 – Mapping of the electrodes receptive fields.

8 Conclusion

This internship enabled me to acquire valuable experience in neuroprosthetics research. Indeed, I developed in-depth knowledge in neural prosthesis, brain machine interfaces and non-human primates experiments. I gained more specific knowledge in unsupervised machine learning algorithms, as well as a better understanding of spike sorting and more generally clustering methods and their challenges. Furthermore, I became more comfortable writing code in Matlab and Bash. Non-human experiments are an essential step in medical research, and it was very interesting to learn about its regulations and procedures. It definitely triggered my curiosity in bioethics. Spending 6 months in the United States helped me improve my skills in English significantly, and discover a new culture. Finally, working in a neuroscience research lab and contributing in such an important project for neuroprosthetics such as the study of artificial feedback like proprioception and the creation of more naturalistic prosthesis was fascinating.

9 Acknowledgements

Foremost, I would like to express my sincere gratitude to Joseph E. O'Doherty, who taught me how to conduct non-human primate experiments, shared with me his knowledge on neuroscience and brain control interfaces, and offered me a valuable help on my spike- sorting algorithm. My sincere thanks also goes to Pr. Philip N. Sabes, who provided me the opportunity to join his lab, and enabled me to acquire experience in neuroprosthetics research. I would also like to thank Mariana Cardoso and Lindsey Presson, for their continuous support, help and advice during my internship.

10 Bibliography

1. Dadarlat, M. C., O'doherty, J. E., & Sabes, P. N. (2015). A learning-based approach to artificial sensory feedback leads to optimal integration. *Nature neuroscience*, 18(1), 138-144.
2. Yazdan-Shahmorad, A., Diaz-Botia, C., Hanson, T. L., Kharazia, V., Ledochowitsch, P., Maharbiz, M. M., & Sabes, P. N. (2016). A large-scale interface for optogenetic stimulation and recording in nonhuman primates. *Neuron*, 89(5), 927-939.
3. Chaisanguanthum, K. S., Shen, H. H., & Sabes, P. N. (2014). Motor variability arises from a slow random walk in neural state. *Journal of Neuroscience*, 34(36), 12071-12080.
4. Makin, J. G., Dichter, B. K., & Sabes, P. N. (2015). Learning to estimate dynamical state with probabilistic population codes. *PLoS Comput Biol*, 11(11), e1004554.
5. Sainburg, R.L., Poizner, H. & Ghez, C. Loss of proprioception produces deficits in interjoint coordination. *J. Neurophysiol.* 70, 21362147 (1993).
6. Sainburg, R.L., Ghilardi, M.F., Poizner, H. & Ghez, C. Control of limb dynamics in normal subjects and patients without proprioception. *J. Neurophysiol.* 73, 820835 (1995).
7. van Beers, R.J., Sittig, A.C. & Gon, J.J. Integration of proprioceptive and visual position- information: an experimentally supported model. *J. Neurophysiol.* 81, 13551364 (1999).
8. Ernst, M.O. & Banks, M.S. Humans integrate visual and haptic information in a statistically optimal fashion. *Nature* 415, 429-433 (2002).
9. Morgan, M.L., Deangelis, G.C. & Angelaki, D.E. Multisensory integration in macaque visual cortex depends on cue reliability. *Neuron* 59, 662-673 (2008).

10. McGuire, L. M., & Sabes, P. N. (2009). Sensory transformations and the use of multiple reference frames for reach planning. *Nature neuroscience*, 12(8), 1056-1061.
11. Dadarlat, M. C., O-Doherty, J. E., & Sabes, P. N. (2013). A learning-based approach to artificial sensory feedback: intracortical microstimulation (ICMS) replaces and augments vision. In 6th International IEEE EMBS Conference on Neural Engineering.
12. Dadarlat, M. C., & Sabes, P. N. (2016). Encoding and Decoding of Multi-Channel ICMS in Macaque Somatosensory Cortex. *IEEE Transactions on Haptics*, 9(4), 508-514.
13. <http://cti.itc.virginia.edu>
14. https://upload.wikimedia.org/wikipedia/commons/c/c4/1421_Sensory_Homunculus.
15. <https://www.technologyreview.com/s/542581/paralyzed-mans-arm-wired-to-receive-brain-signals/>
16. O'doherty, J. E., & Sabes, P. N (2017). Analysis of electrical stimulation artifacts and their removal from bidirectional neural interfaces (paper in publication process)
17. Watson, A. B., & Pelli, D. G. (1983). QUEST: A Bayesian adaptive psychometric method. *Attention, Perception, & Psychophysics*, 33(2), 113-120.
18. Rey, H. G., Pedreira, C., & Quiroga, R. Q. (2015). Past, present and future of spike sorting techniques. *Brain research bulletin*, 119, 106-117.
19. Magland J.F., Jason E Chung J.E. & al., A fully automated approach to spike sorting (2017) (paper in publication process)
20. Hartigan, J. A., & Hartigan, P. M. (1985). The dip test of unimodality. *The Annals of Statistics*, 70-84.
21. Elbatta, M. T., & Ashour, W. M. (2013). A dynamic method for discovering density varied clusters. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 6(1), 123-134.
22. Gibson, S., Judy, J. W., & Markovi-, D. (2012). Spike sorting: The first step in decoding the brain: The first step in decoding the brain. *IEEE Signal processing magazine*, 29(1), 124- 143.
23. L. S. Smith and N. Mtetwa, -A tool for synthesizing spike trains with realistic interference,- *J. Neurosci. Methods*, vol. 159, no. 1, pp. 170-180, 2007.
24. J. Martinez, C. Pedreira, M. J. Ison, and R. Quian Quiroga, -Realistic simulation of extracellular recordings.- *J. Neurosci. Methods*, vol. 184, no. 2, pp. 285- 293, Nov. 2009.
25. C. Pouzat, O. Mazor, and G. Laurent, -Using noise signature to optimize spike-sorting and to assess neuronal classification quality,- *J. Neurosci. Methods*, vol. 122, no. 1, pp. 43-57, 2002.
26. A. F. Atiya, -Recognition of multiunit neural signals,- *IEEE Trans. Biomed. Eng.*, vol. 39, no. 7, pp. 723-729, 1992.
27. S. Takahashi, Y. Anzai, and Y. Sakurai, -Automatic sorting for multi-neuronal activity recorded with tetrodes in the presence of overlapping spikes,- *J. Neuro- physiol.*, vol. 89, no. 4, pp. 2245-2258, Apr. 2003.
28. Oliveira, L. M., & Dimitrov, D. (2008). Surgical techniques for chronic implantation of microwire arrays in rodents and primates. *Methods for Neural Ensemble Recordings*, 21-46.
29. Bernert, Marie, and Blaise Yvert. "Fully unsupervised online spike sorting based on an artificial spiking neural network." *BioRxiv* (2017): 236224.