République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur Et de La Recherche Scientifique

Université de Ghardaïa

N° d'ordre :
N° de série :

Faculté des Sciences et Technologies
Département des Sciences et Technologie

Mémoire présenté en vue de l'obtention du diplôme de

MASTER

Domaine : *Sciences et Technologies*
Filière : *Automatiques*
Spécialité : *Automatiques et systèmes*

Par Nawel RAI, Djalila BEN KHELIFA

Thème

Online Spike Sorting - Application on Real Data Electrocorticography (ECoG)

Soutenu publiquement le : 18/06/2018

Devant le jury :

Mr. Hakim LAKDARI	MAA	Univ. Ghardaïa	**Président**
Mr. Mohammed ARIF	MAA	Univ. Ghardaïa	**Examinateur**
Mr. Boumedean LAADJAL	MAA	Univ. Ghardaïa	**Examinateur**
Mr. Abdessalam KIFOUCHE	MAA	Univ. Ghardaïa	**Encadreur**

Année universitaire 2017/2018

Dedication

To my precious family,

My mother,

My father,

My sisters,

My brothers,

My big family

To my partner

To my special friends every one with his name,

To CAST,

I dedicate this work to you...

DJALILA

Dedication

I dedicate this memoir:

To My great parents

My mother, who taught me that even the largest task can be accomplished if it is done one step at a time.

My father, who taught me that the best kind of knowledge to have is that which is learned for its own sake.

To those who are the source of my inspiration and my courage, to whom I owe love and gratitude:

To my brother Laid and my sister Feryal

To my dear cousins: Amel, Razika and Narimene

To my partner Djalila BENKHELIFA.

Nawel

ACKNOWLEGMENTS

As a preamble to this memoir we thank ALLAH, the all-powerful and merciful one who helps us and who has given us the strength and patience to accomplish this modest work.

This memoir would not have been possible without the conscious intervention of a large number of people.

We would first like to thank Mr. Abdessalam KIFOUCHE very warmly for allowing us to benefit from his supervision. The advice he gave us patience, the confidence he has shown us were decisive in carrying out our research work.

We would like to express our sincere thanks to all the teachers of the Faculty of Science and Technology who have taught us and who through their skills have supported us in the pursuit of our studies.

Our heartfelt thanks also go to the members of the jury Mr.ARIF, Mr.LAADJAL and Mr.LAKDARI for their interest in our research by agreeing to examine our work and to enrich it with their proposals.

Our warmest thanks go to B. Mohcen, R. Kamel and H. Djamel and all CAST members for their help and encouragement

Finally, we would not dare to forget to thank our families and our friends who by their prayers and their encouragement, we were able to overcome all the obstacles.

ABSTRACT

Spike sorting technique is one of the Brain Machine Interface's major steps .This technique is used in electrophysiological data analyzing, it depends on the electrical recorded neuronal activity detected by electrodes placed on the motor cortex. SS process has four main steps; filtering, detection, feature extraction and classification. In this work, using real electrocorticography dataset extracted from experiment on rats' motor cortex; we aim to the online spike sorting, in other word, SS process in done simultaneously during this experiment.

***Key words**:* Spike sorting, Brain Machine Interface, electrophysiological data analyzing, neuronal activity, electrodes, electrocorticography, motor cortex, filtering, detection, feature extraction, classification, online spike sorting.

ملخـــــــص

تقنية فرز السبايك هي واحدة من الخطوات الرئيسية لواجهة آلة-الدماغ. ويستخدم هذا الأسلوب في تحليل البيانات الكهربأئية، ويعتمد ذلك على النشاط العصبي المسجل بالكهرباء المكتشفة بواسطة الأقطاب الكهربائية الموضوعة على القشرة الحركية. عملية الفرز هذه لديها أربع خطوات رئيسية؛ التصفية، الكشف، استخراج المعالم والتصنيف. في هذا العمل، باستخدام مجموعة بيانات تخطيط كهربية قشرة دماغ حقيقية مستخرجة من التجارب على القشرة الحركية للجرذ؛ نحن نهدف إلى فرز آني، بتعبير آخر، عملية الفرز تكو في نفس الوقت خلال هذا الاختبار.

كلمات مفتاحية: تقنية فرزالسبايك، واجهة آلة-الدماغ، تحليل بيانات الكهربية، النشاط العصبي، الأقطاب الكهربائية، القشرة الحركية، التصفية، الكشف، استخراج المعالم، التصنيف، تخطيط كهربية قشر الدماغ، فرز آني.

Résumé

La technique de Séparation des impulsions est l'une des étapes majeures de l'Interface Cerveau Machine. Cette technique est utilisée dans l'analyse des données électro-physiologiques, elle dépend de l'activité neuronale enregistrée électriquement détectée par les électrodes placées sur le cortex moteur. Le processus de séparation des impulsions comporte quatre étapes principales ; filtrage, détection, extraction de caractéristiques et classification. Dans ce travail, en utilisant la base de données electrocorticography réel extrait de l'expérience sur le cortex moteur des rats; nous visons la séparation en ligne, en d'autres termes, ce processus est fait simultanément pendant cette expérience.

Mots clé : Séparation des impulsions, l'Interface Cerveau Machine, analyse des données électro-physiologiques, activité neuronale, électrodes, cortex moteur, filtrage, détection, extraction de caractéristiques, classification, electrocorticography, séparation en ligne.

Summary

Figures List

Tables List

Abbreviations List

BMI	Brain Machine Interface
BCI	Brain Computer Interface
MMI	Mind Machine Interface
EEG	Electroencephatography
DOC	Disorders Of Consciousness
VEP	Visual Evoked Potential
EP	Evoked Potential
ERP	Event Related Potential
ECoG	Electrocorticography
MEG	Magnet encephalography
SQUID	Superconducting Quantum Interface Device
PET	Position Emission Tomography
FMRI	Functional Magnetic Resonance Imaging
FNIR	Functional Near Infra-Red
NEO	Nonlinear Energy Operator
TEO	Teager Energy Operator
PCA	principal Component Analysis
SSP	Spike Sorting Process
FE	Feature Extraction
SNR	Signal to Noise Ratio
SPC	Superparamagnetic Clustering
BC	Bayesian Clustering
TM	Template Matching
MF	Matched Filter
AV	Absolut Value

GENERAL INTRODUCTION

General Introduction

As the power of modern computers grows alongside our understanding of the human brain, we have become more capable of making the human dream, to interact with the computer through the activities of the brain, which was a long standing dream for the scientists by the advent of science fiction movies using advanced electronic devices to capture brain signals and control real world devices. As for physically disabled people, they face difficulties in equating their life with everybody else's, and that's a big issue in every society today. The scientists have begun using intra\extracellular recording technics, these technics were a fantasy thing in medical technologies for the treatment of disorders such as paralysis, epilepsy, and even cognitive and memory loss. Many of these technologies are based upon the idea of brain machine interfaces (BMIs), in which implanted electronics record and decode brain signals that can be used to control machines such as computers or prosthetic limbs. The simplest definition of BMI (Brain Machine Interface) is the way for making a direct communication between the brain and the external world.

The most common method for measuring neural firing is direct electrical recording, which uses electrodes to record the voltage patterns of nearby neurons. Each neuron yields a characteristic electrical signal, known as a spike but from the recording step, we'll obtain a superposition of many signals from the flow of current coming from the neighbor neurons, and as we know about the information is supported in the time distance between individual neurons firing. Thus, spike sorting; the grouping of spikes by shape; can be used to match measured signals to their generating neurons. More precisely, spike sorting extracts the firing times and corresponding neuron labels of noisy electrophysiological recordings.

In this thesis, we will go through three linked chapters. As an entry, we will talk about the Brain Machine Interface principals and types, its different applications and signal recording method. After that, we will learn about spike sorting, specifically, online sorting technique and its main four steps: signal filtering, spikes detection and alignment, features extraction and spikes classification and clustering. Finally, using MATLAB software, we will present the obtained results of online spike sorting applied on real ECoG data extracted during the experiment on rats' behaviors and activities.

CHAPTER 1 : BRAIN MACHINE INTERFACE

1 Introduction

One of the most important and distinguishing aspects of humans is the ability to communicate. Communication between people is richer and more complex than any other form of communication, and plays a vital role in any relationship. Similarly, as artificial devices become more complicated and play a rapidly waxing role in everyday life, communicating effectively with them becomes increasingly important.

It is impossible to directly convey thoughts, emotions, or concepts between people. Instead, these must be translated into verbal or written statements, gesticulations, facial gestures, drawings, or other recognizable expressions using a new technology that is Brain Machine Interface (BMI)

BMI gives the paralyzed people another way to communicate with the outside world. In addition, these interfaces do not only have medical virtues, they could also be applied in many other areas.

Currently, the latest advances have resulted in this new technology that can lead us to a better world. So, the BMI can revolutionize the modern world.

2 Brain Machine Interface

Brain Machine Interface (BMI), often called Brain Computer Interface (BCI) or sometimes called Mind Machine Interface (MMI**).**It is the way for making direct communication pathway between the brain and an external device. It allows us to transfer and use information from distinct brain states for communicating with a machine.

Brain Machine Interface (BMIs) started with Hans Berger's experiment of brain electrical signals recording applied on rabbits witch is actually referenced electroencephalography (EEG). In 1924 Berger recorded an EEG signals from a human brain for the first time. By analyzing EEG signals Berger was able to identify oscillatory activity in the brain, such as the alpha wave (8–12 Hz), also known as Berger's wave. The first recording device used by Berger was very elementary, which was in the early stages of development, and was required to insert silver wires under the scalp of the patients. In later stages, those were replaced by silver foils that were attached to the patients head by rubber bandages later on Berger connected these sensors to a Lippmann capillary electrometer, with disappointing results.

More sophisticated measuring devices such as the Siemens double-coil recording galvanometer, which displayed electric voltages as small as one ten thousandth of a volt, led to success. Berger analyzed the interrelation of alternations in his EEG wave diagrams with brain diseases. EEGs permitted completely new possibilities for the research of human brain activities[1, 2].

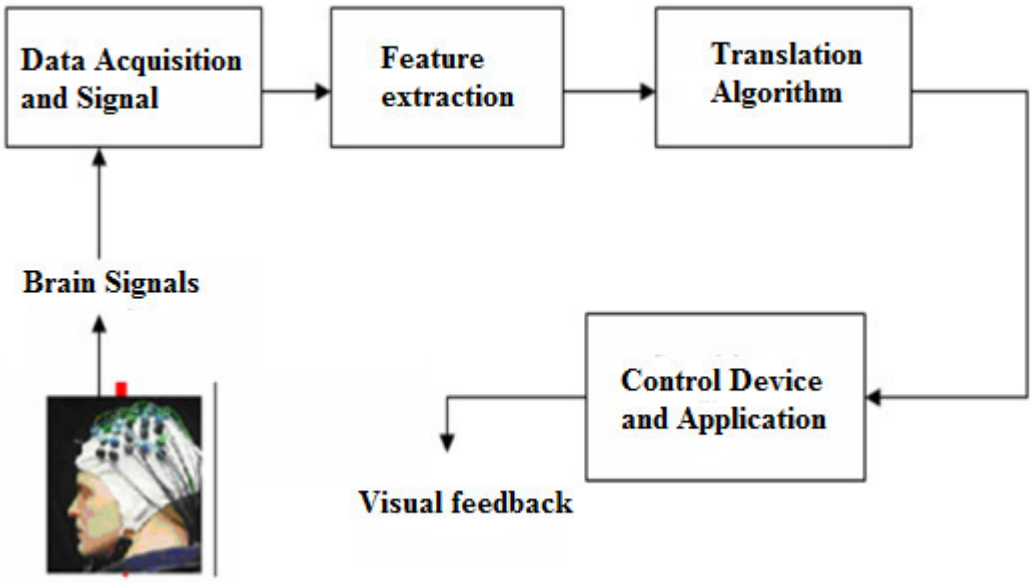


Figure 1.1: Representation of a BMI

3 Human Brain

Amazingly, nothing in the world can be compared with the human brain. The brain is undoubtedly the most complex organ found among the carbon-based life forms. So complex it is that we have only vague information about how it works.

The most relevant part of brain concerning BMI‘s is the cerebral cortex. Cerebral cortex is responsible for many higher order functions like problem solving, language comprehension and processing of complex visual information.

The brain produces electrical signals, which, together with chemical reactions, let the parts of the body communicate.

Nerves send these signals throughout the body. Brain cells include neurons and glial cells. The brain is part of the central nervous system (CNS), which consists of large brain, little brain (or cerebellum), brainstem and spinal cord

The brain is connected to the spinal cord, which runs from the neck to the hip area. The spinal cord carries nerve messages between the brain and the body. The nerves that connect the CNS to the rest of the body are called the peripheral nervous system. Finally, the autonomic nervous system controls our life support systems that we don't consciously control, like breathing, digesting food, blood circulation, etc [3, 4]

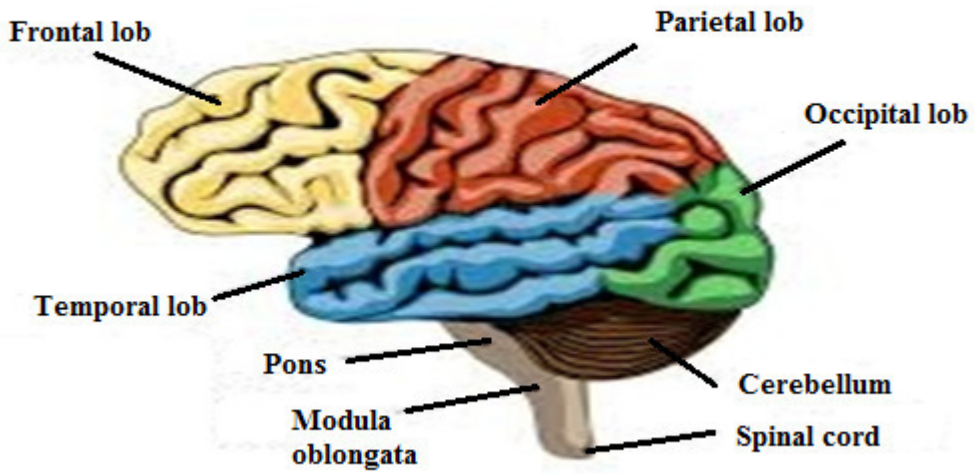


Figure 1.2: Human brain

4 BMI Components

A brain-machine interface (BMI) is a combination of several hardware and software components. The hardware consists of an EEG machine and a number of electrodes scattered over the subject‘s skull. The EEG machine -that is connected to the electrodes via thin wires- records the brain-electrical activity of the subject. The software system has to read, digitize and preprocess the EEG data [3].

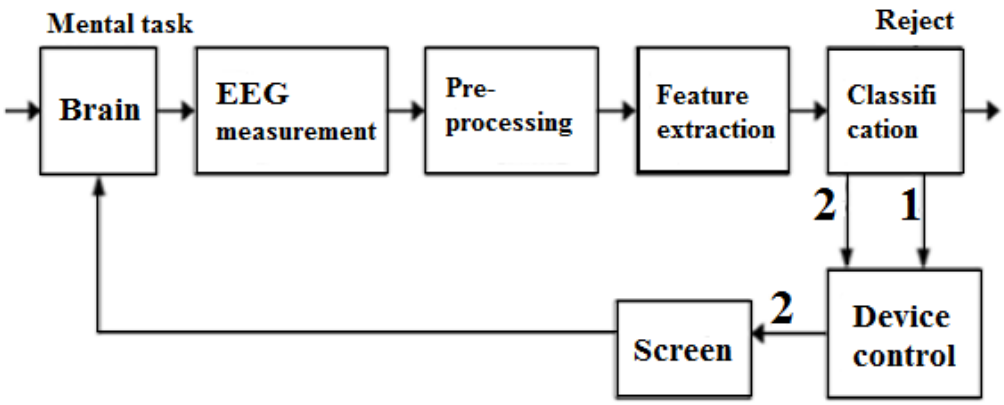


Figure 1.3 :A BMI based on the classification of two mental tasks. The user is thinking task number 2 and the BCI classifies it correctly and provides feedback in the form of cursor movement [3].

The BMI components can be described as follow:

4.1 Implant Device

The EEG is recorded with electrodes, which are placed on the scalp. Electrodes are small plates, which conduct electricity. They provide the electrical contact between the skin and the EEG recording apparatus by transforming the ionic current on the skin to the electrical current in the wires. To improve the stability of the signal, the outer layer of the skin called stratum corneum should be at least partly removed under the electrode. Electrolyte gel is applied between the electrode and the skin in order to provide good electrical contact. [3].

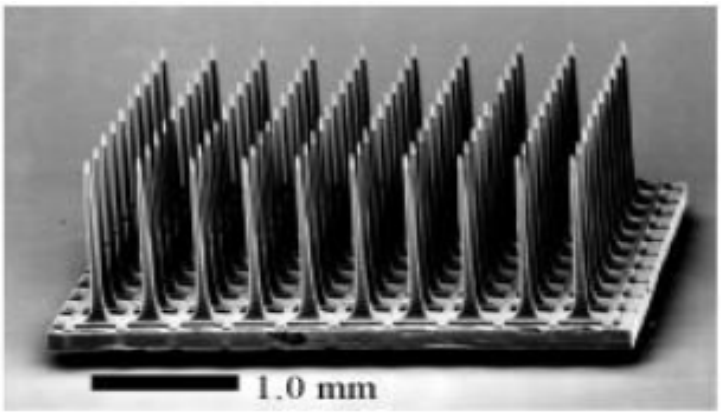


Figure 1.4 : An array of microelectrodes[8]

4.2 Signal Processing Section

4.2.1 Multichannel Acquisition Systems

Electrodes interface directly to the non-inverting opamp inputs on each channel. At this section amplification, initial filtering of EEG signal and possible artifact removal takes place [3].

4.2.2 Spike Detection

Real time spike detection is an important requirement for developing brain machine interfaces. Incorporating spike detection will allow the BMI to transmit only the action potential waveforms and their respective arrival times instead of the sparse, raw signal in its entirety [3].

4.2.3 Signal Analysis

Feature extraction and classification of EEG are dealt in this section. In this stage, certain features are extracted from the preprocessed and digitized EEG signal. In the simplest form a certain frequency range is selected and the amplitude relative to some reference level measured [3].

4.3 External Device

The classifier‘s output is the input for the device control. The device control simply transforms the classification to a particular action. The action can be, e.g., an up or down movement of a cursor on the feedback screen or a selection of a letter in a writing application. However, if the classification was “nothing” or “reject‖”, no action is performed, although the user may be informed about the rejection. It is the device that subject produce and control motion. Examples are robotic arm, thought controlled wheel chair etc [3].

4.4 Feedback

Real-time feedback can dramatically improve the performance of a brain–machine interface. Feedback is needed for learning and for control. Real-time feedback can dramatically improve the performance of a brain–machine interface. In the brain, feedback normally allows for two corrective mechanisms. One is the online’ control and correction of

errors during the execution of a movement. The other is learning: the gradual adaptation of motor commands, which takes place after the execution of one or more movements.

In the BMIs based on the operant conditioning approach, feedback training is essential for the user to acquire the control of his or her EEG response. The BMIs based on the pattern recognition approach and using mental tasks do not definitely require feedback training. However, feedback can speed up the learning process and improve performance. Cursor control has been the most popular type of feedback in BMIs. Feedback can have many different effects, some of them beneficial and some harmful. Feedback used in BMIs has similarities with biofeedback, especially EEG.[3]

5 BMI Function

Applications of Brain Computer Interface base its functionality on either observing the user state or allowing the user to deliver his/her ideas. BCI system records the brain waves and sends them to the computer system to complete the intended task. The transmitted waves are the refor used to express an idea or control an object [3].

5.1 Communication and Control

Brain Machine interface (BMI) systems build a communication bridge between human brain and the external world eliminating the need for typical information delivery methods. They manage the sending of messages from human brains and decoding their silent thoughts. Thus they can help handicapped people to tell and write down their opinions and ideas via variety of methods such as in spelling applications semantic categorization, or silent speech communication

BMIs can also facilitate hands-free applications bringing the ease and comfort to human beings through mind-controlling of machines. They only require incorporating brain signals in order to accomplish a set of commands and no muscles intervention is needed. BCI assistive robots can offer support for disabled users in daily and professional life, increasing their cooperation in building their community [5].

5.2 User State Monitoring

Early BMI applications have targeted disabled users who have mobility or speaking issues. Their aim was to provide an alternative communication channel for those users. But

later on, BMI enters the world of healthy people as well. It works as a physiological measuring tool that retrieves and uses information about an individual's emotional, cognitive or effectiveness state. The target of brain signals utilization has been extended beyond controlling some object or offering a substitution for specific unctions, in what is called passive BMI .According to Garcia-Molina et al. , the precise awareness of the curren temotional or cognitive state can affect the recognition of the mental task associated with the recorded brain waves. Another beneficial employment of such information is to determine the state itself and use that knowledge for enhancing various BMI systems. BMI user state monitoring function is considered a helpful hand in Human Computer Interfaces and adapts them according to the estimated user emotional or cognitive state. It participates in a shared control environment and decides the best type of control that might be used in certain situations.

It also contributes in the development of smart environments and emotion controlling applications. Working conditions' assessment and educational methods' evaluation are examples of other fields that could benefit from measuring user's brain state. The next section highlights some applications that exploit brain computer interface [5].

6 BMI Applications

BMI is interesting area to researchers because it can solve many problems which seem to be impossible. The essential target of BMI applications is to convert the user's intent or thoughts to an action in external device or computer and control to these devices. Many applications of BMI concerned on patients suffer from disorders of consciousness (DOC).These patients unable to make communication with their around world. By using BMI, these patients can control some devices to perform basic and important jobs they need without helping like moving with wheelchair, getting something for eating or drinking by using robotic legs or arms controlled by brain.

BMI technologies are used to restore the vision to blinds by connecting an external camera with brain. Applications on device control not include patients only, but also healthy users like whose needs to perform many jobs at the same time like divers, astronauts and drivers where they keep their hands on swimming, operate equipment and the steering wheel .Rabie et al.developed a BMI based system that can help disabled persons to use the web through their brains only. The authors developed a technique that captures the eye signals through the brain to select the appropriate letters as well as words to be written on the web

browser. Another application that has been developed is the wheelchair simulator that is controlled also by the BMI signals.

BMI used also on User-state monitoring which make alert to sleepy drivers or students. Also, it extended physically to measure the heart beats for users. Many applications focused in entertainment and playing games especially after using 3D monitors, certain glasses and an EEG headset where the control on the game by thoughts. EEG combined sometimes with eye movement on some applications for security and safety where the system can monitor suspicious objects, deviant behavior or arousal state. A common BMI application is neurofeedback training to improve working, attention, executive functions and memory. Neuroergonomics is an evaluation application used to estimate how well human abilities match a technology. BMI used also in education and training techniques. By sing BMI based on EEG, patient can control or move the cursor by mental thoughts where the patient can select words or letters [6].

Figure 1. 5: Wheel chair controlled by brain signals [7].

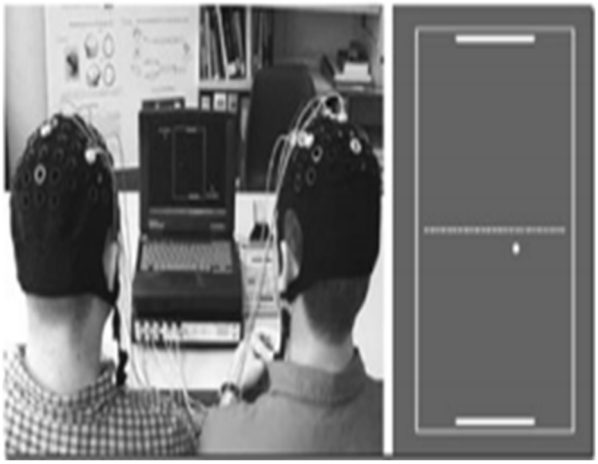


Figure 1.6: Computer game Pong for two players [8].

7 BMI Classification

BMIs have been categorized into four (04) classes types in recent years, these classes are discussed below:

7.1 Brain Signals

Numerous recording techniques make it possible to capture brain activity directly or indirectly and to convert it into signals that can be used to perform a BMI. Some of these techniques, called invasive require the surgical implantation of electrodes under the skull; to measure the electric field resulting from neuronal activities. There is a wider issue concerning the term ‘invasive’. In biomedical engineering, any technology that deposits any eternal elements on the sub-epidermal tissue is considered invasive. Other types measuring brain activity by a set of electrodes placed on the cortical surface called non-invasive [9].

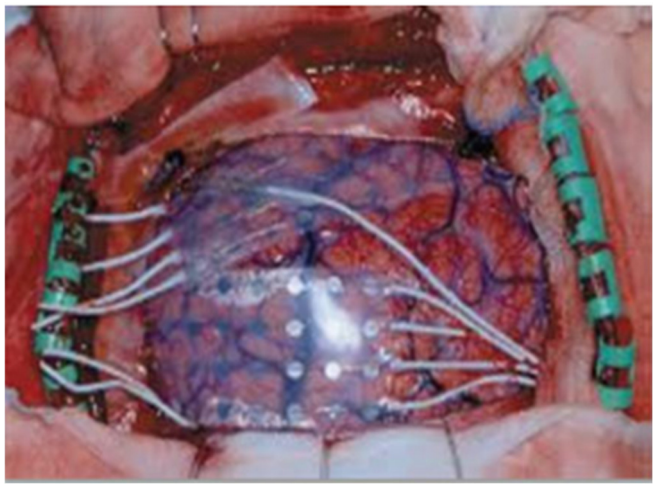


Figure 1.7: Invasive BMI Electrodes [6].

Figure 1.8: Wireless noninvasive signal capturing device [6].

7.2 Synchronous and Asynchronous

BMI system is called a synchronous when the user interaction with the system is done at certain period of time. In other words, the system has to impose the subject to interact with it at certain period of time. Otherwise, the system will not be able to receive the subject signals. On the other hand, in asynchronous BMI, also named as “self-paced”, the subject is able to perform its mental tasks at any period of time and the system will react to the mental activities. Therefore, the subject is free to have the activity at any period of time [10].

7.3 Dependent and Non-Independent

The BMI machines have been classified as Dependent and Independent. The first one, the dependent BMIs does not employ the usual ways of brain output to transport and dispatch the relevant message. A dependent BMI requires the presence but does not use normal output

pathway to produce the brain signals that feed the interfacing computer. For instance, let us consider a matrix of alphabet letters presented in a video display, which are sequentially illuminated; the human subject is trained to choose the letter he/she wants just via ocular fixation .This act produces a Visual Evoked Potential (VEP) on the occipital scalp significantly larger than the responses provoked by the other flashing letters which are not fixated by the Patient. Therefore, the relevant signal is coming from the EEG, but it is due to the sight focus direction.

An Independent BMI, on the other hand, is entirely free from the physiological output pathways of brain as the relevant signal is not generated by propagating signals along peripheral nerves, muscles, or other physiological outputs [4].

7.4 Spontaneous and Evoked and Event-Related

Evoked potentials (EPs) appear in the brain as a result of a particular stimulus, e.g., a flashing letter, whether the subject is interested in it or not. EPs are time locked to the stimulus. Other brain signals can be completely spontaneous, such as those related to movement intentions in the sensory motor cortex and are thus not a result of specific input. Finally, a third class of signals is dubbed ‘event-related potentials’ (ERP). These are related to evoked potentials but include brain responses that are not directly elicited by the stimulus; they can include cognitive signals, among other psychological manifestations In fact the term ERP is seen as a more accurate term for all but the most restricted simulation protocols. It is thus the preferred term instead of EP [4].

8 Neuromagink Methods

8.1 Electroencephalography (EEG)

Many brain activities could be generated by potential actions of the subject or by changing in the blood flow. Recording such activities could be done directly by monitoring electrophysiological signals. The most used methods are [10]:

Electroencephalography (EEG) consists of brain electrical activity patterns as measured from the outside of the skull. The first observation of such patterns in animals was made by Richard Caton, in Liverpool, in 1875(Cooper et al., 1969). Caton was studying brain activity in cats, monkeys and rabbits using cortical non-polarizable electrodes connected to a galvanometer when he observed that “feeble currents of varying direction pass through the

multiplier [instrumentation] when two electrodes are placed on two points of the external surface". In 1876, Danilevskey observed a change in cortical potentials due to acoustic stimuli.

It was not until 1929 that similar studies were reported in humans. At that time, Hans Berger (Figure 1.9.A), in Germany, reported on a study using platinum wires pushed into the scalp, zinc-plated steel needles, and various other metals in an attempt to observe electroencephalographic activity in humans. Berger also gave electroencephalography its name. He is widely considered to be the inventor of EEG. Later, after 1930, when the galvanometer was replaced by valve amplifiers and a.c. coupling, better technology was available. One of the most important developments was the multi-channel EEG system developed by Grey Walter, which he called the EEG toposcope (Figure 1.9.B). This device allowed for studies of brain potentials and their temporal relationships. It is a very useful tool and inspired the multi-channel devices used today. Meanwhile, Edgard Adrian and others continued Berger's work. In the 50's, after the invention of the transistor, researchers returned to using d.c. instrumentation. Since then, most developments have been made on making the instrumentation less prone to noise and developing better electrodes [4].

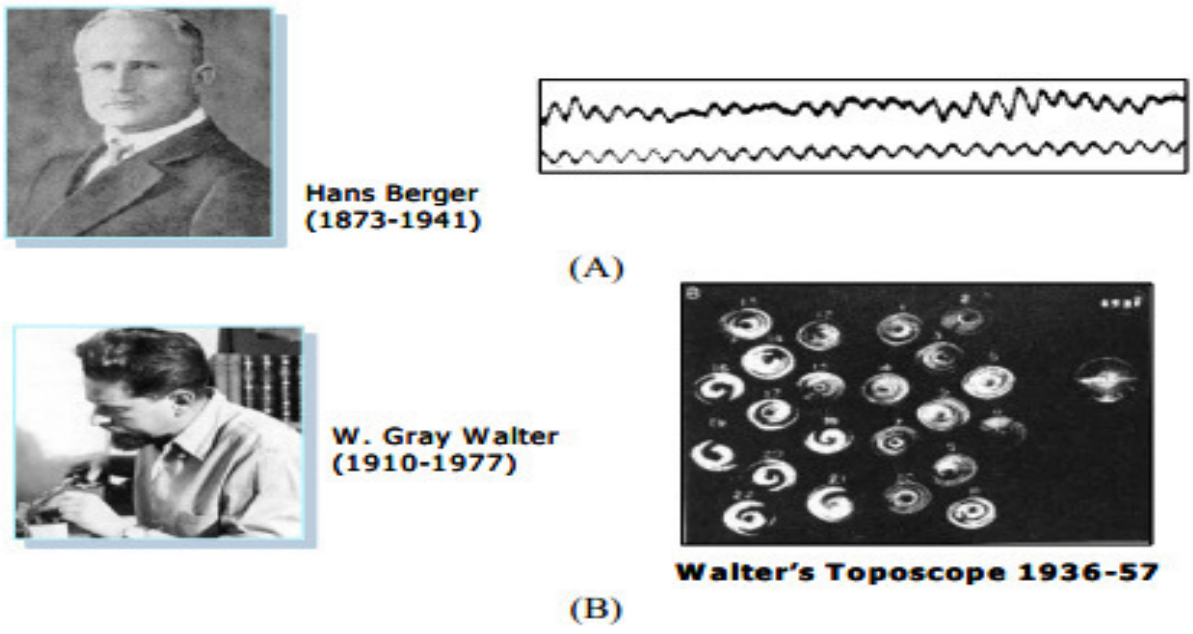


Figure 1.9: Some of the pioneers in the development of EEG technology. A) Hans Berger and the first reported human EEG signals. B) Grey Walter and the EEG toposcope.

EEG measures the electrical activity by detecting the electrical potential difference between a point on the scalp and a ground. More specifically, EEG mostly measures the potentials resulting from current flow during excitation of the dendrites of pyramidalneurons in the cerebral cortex. Other, deeper cells may contribute to the detected signal as well, but this is usually a small affect.

The amplitude of the potentials measured on the neural cells' membranes is in the order of several tens of mV. However, the tissues between the electrode and the monitored cells, and the distance between the neurons and the electrodes, cause the signal to be attenuated by many orders of magnitude. Thus, the activity of a few neural cells cannot be detected through EEG. Instead, EEG is a result of joint activity of thousands of underlying neurons activated together (so called synchronous activity). The amplitude of the EEG signal is proportional to the number of synchronously activated neurons and the size of the synchronous area. Even so, at besEEG amplitudes are usually in the order of 100 microvolt or less [4].

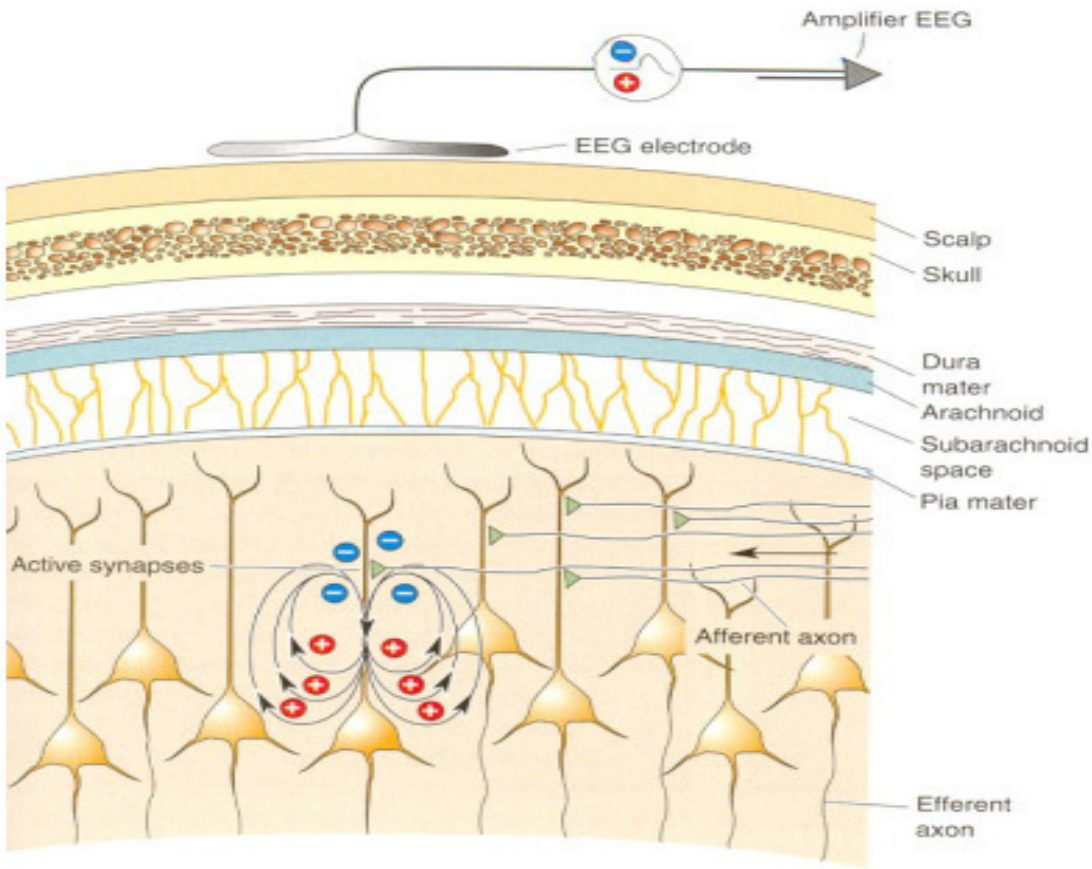


Figure 1.10: Generation of cortical potentials that are measured from the skull using EEG instrumentation.

8.2 Electrocorticography (ECoG)

Electrocorticography (ECoG) is used to measure the electrical activities of the brain through an invasive procedure. In other words, the skull of the subject has to be removed and the electrodes are placed directly on the service of the brain.

Therefore, since the electrodes are placed directly on the skull, the special resolution of the measured signals are much better than EEG and signal-to-noise ratio is superior due to greater vicinity to neural activity. However, the usage of ECoG is very limited to the exposed brain area and it is almost impossible to be used outside of a surgery room [10].

8.3 Magnetoencephalography (Meg)

Magnetoencephalography (MEG) is used to identify and analyze the magnetic field of the brain using a functional neuroimaging technique It was first invited by David Choen in 1968.[15] It utilizes a superconducting quantum interference device (SQUID) that is extremely sensitive to the magnetic disturbances created during neuronal activity. This device can be used to non-invasively detect the magnetic field signals around the scalp (50–500 fT) that are generated by neural activity. Modern MEG devices typically use helmet-shaped sensor arrays of more than 300 SQUIDs that are systematically arranged to cover the entire scalp [28]. Magnetoencephalography (MEG) devices are still too bulky to become a convenient BMI modality for everyday use.

8.4 Positron Emission Tomography (Pet)

Positron Emission Tomography (PET) is used to observe metabolic processes in the body and it is similar to SPECT; however, in PET a pair of gamma rays is emitted due to radionuclides injection in the patients. In other words, this radionuclides emits positrons that interacts with the electrons located in the monitored/canned area. This interaction generates the gamma rays. Using these gamma rays, an image can be constructed. Unfortunately PET has high operating cost which makes it not preferable to be used [10].

8.5 Functional Magnetic Resonance Imaging (Fmri)

Functional magnetic resonance (fMRI, Heeger&Ress 2002) is a noninvasive method of measuring neuronal activity in the human brain. FMRI detects changes in the concentration of deoxyhemoglobin, dependent on a complex interplay among blood flow, blood volume and

cerebral oxygen consumption (Heeger et al., 2000; Heeger and Ress, 2002). When neurons increase their activity with respect to a baseline level, a modulation of the deoxyhemoglobin concentration is induced, generating the so-called blood oxygen level dependent (BOLD) contrast (Boynton et al., 1996). BOLD dynamics is characterised by an initial transient small decrease below baseline due to initial oxygen consumption (negative dip), followed by a large increase above baseline, due to an oversupply of oxygenated blood only partially compensated by an increase in the deoxygenated venous blood volume [4].

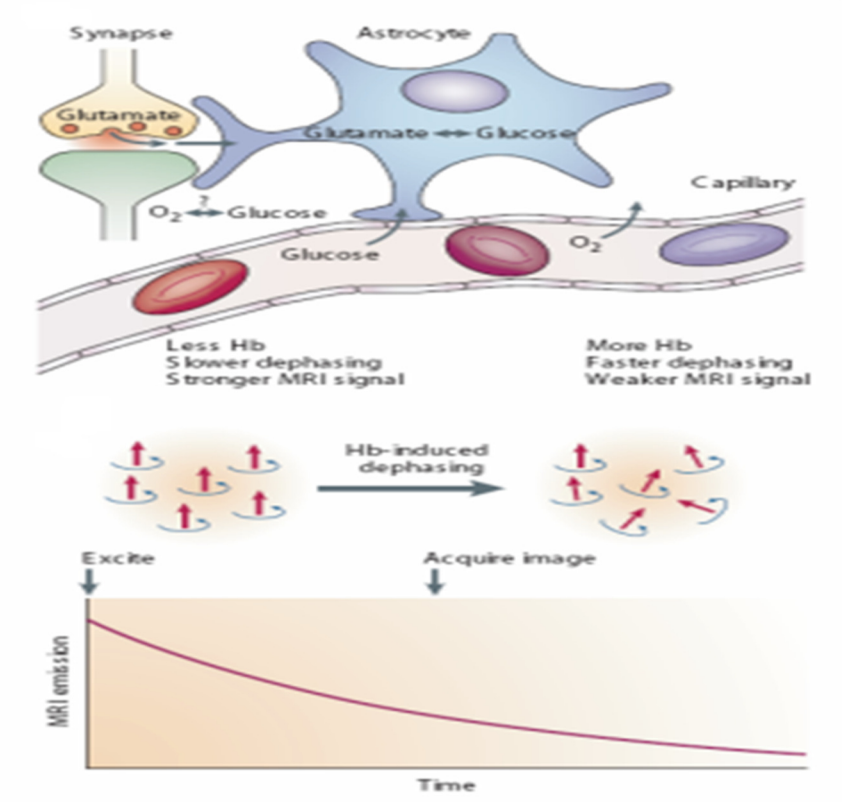


Figure 1.11: BOLD signal generation [4].

8.6 Optical Imaging (Functional Near Infrared (FNIR))

fNIRS technology projects the infrared light into the brain to measure the changes at various wavelengths as the light is reflected back out. Usually, the fNIRS detects the localized blood volume and oxygenation changes. In fact, fNIRS is used to shape the function maps of the brain activities since the fluctuations in tissue oxygenation modulate the scattering and absorption of the infrared light photons to varying amounts. Therefore, images similar to the

traditional Functional Magnetic Resonance Imaging could be generated with high spatial resolution (<1 cm) at the expense of lower temporal resolution (>2–5 seconds). Unfortunately, due to the low temporal resolution, fNIRS is not preferred to be used by most of the researchers [28].

9 Conclusion

The BCI reads the waves produced from the brain at different locations in the human head, translates these signals into actions, and commands that can control the computer(s). The field of BCI is one of the important fields that deal with brain activities. It is expected that BCI applications will have great effect on our daily life. This chapter focuses on defining the BMI. And also present its different types and applications .There are a plethora of signals, which can be used for BCI. These signals divide into two classes: field potentials and spike. In order to better understand how the brain works we present in the next chapter a necessary step in the brain machine interface that is spike sorting.

CHAPTER 2: ONLINE SPIKE SORTING

1 Introduction

Neurons communicate through electrophysiological signals, which may be recorded using electrodes inserted into living tissue. When a neuron emits a signal, it is referred to as a spike, and an electrode can detect these from multiple neurons.

The point process component of intracellular recording results from the spiking activity of neurons in a background of physical and biological noise. When a recording electrode measures action potentials from multiple cells, these contributions must be disentangled from the background noise and from each other before the activity of individual neurons can be analyzed. This procedure of estimating one or more single cell point processes from a noisy time series is known as spike sorting.

2 Stat Of The Art

In a traditional neural recording system, electrodes provide the direct interface to the brain, and the unamplified raw data is sent outside the body through wires to the rest of the data acquisition hardware. The raw data is recorded onto computer hard disks [11].

Many different approaches have already been tested and introduced in order to address the problem at hand. In terms of spike detection, the most common methods use simple thresholding techniques to extract the actual spike signals from the noisy measurement data. More elaborated algorithms further add specific signal transformations, e.g. using the nonlinear or Teager energy operator (NEO or TEO) (Kim and Kim, 2000; Choi et al., 2006) to the thresholding process in order to decrease the influence of noise or low frequency signal artifacts. Another approach presented by Hulata et al. (2002) uses Wavelet-based decomposition to discriminate actual spike shapes from noise. Such extended detection methods usually yield a more precise detection result, which facilitates the actual sorting process, but at the expense of a higher computational complexity. The most common method of feature extraction and reduction is still the principal component analysis (PCA) (Wood et al.,2004; Wang et al., 2006; Biffi et al., 2008).However, Wavelet or Wavelet packet-based features have also become quite popular as the resulting coefficients are also capable of describing the differences of various spike signals (Oweiss and Anderson, 2002; Hulata et al., 2002; Quiroga et al., 2004). A drawback of this complex method is the necessity of an additional selection step to identify the coefficients that most precisely discriminate between

the various spike shapes. In some cases, geometric features are also obtained from the measured spike events (Fee et al., 1996; Vogelstein et al., 2004) as an alternative or, more commonly, a complement to standard PCA features. The variety of the used spike sorting features indicates that no optimal feature set has yet been established. This might be due to the fact that it is questionable whether a fixed set of features exists that can generally guarantee a suitable spike shape representation. While an analysis of common spike sorting features already shows certain diversity, this is even more apparent in the area of possible classification algorithms. Many different pattern recognition approaches, ranging from simple clustering techniques such as k-means (Hulata et al., 2002; Sato et al., 2007), more elaborate expectation maximization (Kim and Kim, 2003; Wood et al., 2004) and super-paramagnetic clustering methods (Quiroga et al., 2004), to trainable classification algorithms such as neural networks (Kim and Kim, 2000) and support vector machines (SVMs) (Vogelsteinet al., 2004), have already been tested for neural spike sorting. However, none of these approaches has been able to provide generally optimal results to be widely established. Whereas trainable algorithms certainly provide a greater potential, their performance still seems to suffer due to a lack of fundamental training data. Unlike in other areas, for example voice recognition or person identification, in which basic training databases already exist, such datasets have not been established for neural patterns, thereby limiting actual algorithm training. In addition, many spike sorting algorithms also include a final template matching step to further refine the classification result (Zhang et al., 2004; Wang et al., 2006; Sato et al., 2007). However, since the result depends on prior template extraction methods, which usually include the already-mentioned clustering techniques, inaccuracies cannot always be compensated. In conclusion, there still seems to be no optimal method that is generally capable of sorting neural spike patterns. One reason for this might be the lack of fundamental test datasets containing a principal spectrum of theoretically possible spike shapes that could be used to evaluate the different classification algorithms. As this problem also prevents a thorough evaluation of the suitability of different spike features, it is generally difficult to evaluate the feasibility of specific types of features such as principal components or Wavelet coefficients. Although certain studies have shown that Wavelet-based feature extraction potentially outperforms the PCA, this fact cannot be generalized to all possible datasets (Pavlov et al., 2007). Therefore, a priori exclusion of PCA features can limit the ability to discriminate between certain spike forms. In fact, this is also true for other features that might be inferior in most cases but not in general. Hence, it might not be useful to focus on a specific feature extraction technique, as other features might be more suitable on specific occasions. Therefore, in contrast to most

methods, the algorithm presented in this paper regards various possible features, such as principal components and different Wavelet packet coefficients, as well as geometric features. As such an approach requires a unique feature reduction step in order to determine the features most suitable for the particular spikes of the signal, a newly developed feature evaluation step is proposed in this paper. By analyzing the probability distribution of the different feature coefficients, the candidates with the most distinctive multimodal character can be found. Resembling the different spike shape characteristics, this method allows the derivation of spike sorting feature sets that are customized for each individual set of neuronal data [12].

3 Spike Sorting

An important standard in neuroscience is to record intracellular the activity of single neurons with thin electrodes implanted in the brain. Intracellular recordings pick up the spikes of neurons nearby the electrode tip and the job of the experimenter is to determine which spike corresponds to which neuron. This identification is done based on the shape of the spikes, given that, in principle, each neuron fires spikes of a particular shape, depending on the morphology of its dendritic tree and the distance and orientation relative to the recording site, among other factors. Spike sorting is the grouping of the detected spikes into clusters based on the similarity of their shapes. The resulting clusters of spikes correspond to the activity of different putative neurons [13].

The process consists of ordered steps (see Figure2.1), which will be explained in the following sections:

1. Signal filtering
2. Spike detection
3. Feature extraction
4. Spike classification clustering

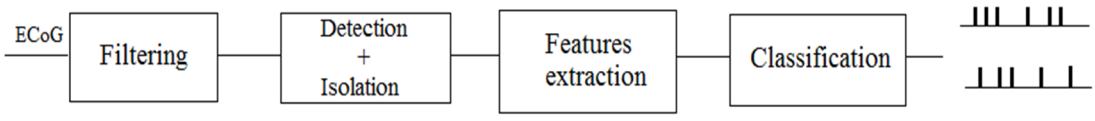


Figure 2.1: Main four steps of spike sorting process.

All of these steps are important for the final result {an efficient and reliable clustering algorithm is of little use if one is unable to extract relevant features from the relevant spikes.

The development of reliable and efficient spike sorting algorithms is lagging behind the recent developments in recording hardware, increasing the importance of improving the algorithms. Hundreds of electrodes can now record signals simultaneously, allowing the recording of many neurons, which helps in providing a bigger picture of the activity. However, being able to classify all the recorded neurons is non-trivial, especially when they are being recorded simultaneously.

A potential problem with neuronal activity recording is making a distinction between one or several close neurons. This is sometimes not possible, and clusters are labeled either as single neurons or multi-units {collections of several neurons which fire together. Many automatic spike sorting algorithms make no distinction between the two, and there are no clear, objective and agreed-upon criteria for making the distinction.

4 Online Spike Sorting

This section presents the proposed solution to the problem of designing an implantable Digital Spike Sorting that implement a Spike Sorting Process. Besides implantable, this processor must be able to transmit, in real-time, the relevant information to the external world through a wireless module as illustrated in Figure 2.2. However, due to the current technologies used in the development of electrode arrays, the SSP is actually a multi-channel system, where multiple parallel channels record the responses of different sets of neurons. Thus, it is highly relevant to provide the means for a single processor to handle multiple channels (preferably all) without missing relevant information, in order to minimize both the implantable device size and its energy consumption. Nonetheless, to ensure a correct operation, such a processor must be limited by the worst-case scenario. For instance, consider the reasonable refractory period (the minimum interval of time between the fires of two spikes from the same neuron) of 5ms and a 16 channel system. The time available to process a spike, from its detection to its classification, is approximately 300 μs (since the interval of time between two spikes from the same neuron has to be enough for processing all the other channels). In this way, once guaranteed the real-time operation, the objective of the proposed work is to increase the number of different channels the dedicated processor is able to process simultaneously within the inherent restrictions.

Figure 2.2: Data transmission to the external world.

To accomplish this objective, the proposed solution is developed in several steps. Each new step represents a lower-level of abstraction as well as contains a more detailed and closer to the hardware description [14].

5 Spike Sorting Steps:

The spike sorting and classification regroup many steps before obtaining sorted signal, first, we have to filter the signal from eliminating noises, than, we have to detect, isolate and align spikes and finally, we cluster spikes using any technique of separation.

5.1 Signal Filtering

The word "filtering" refers to an attempt to extract the important part of some data while eliminating random contributions called "noise" or other unwanted features which obscure the ones that matter. Depending on the applications, we can describe two wide worlds of filters, digital and analogical. If we classify filters on the way of permitted frequencies we can have four types: high-pass, low-pass, band-pass and stop-band.

5.1.1 Filters

Filter is the (i) circuit for analogical signals and (ii) algorithm for digital signals that capable of passing signal from input to output that has frequency within a specified band and attenuating all others outside the band. This is the property of selectivity. There are four basic types of filters depending on the permitted frequencies to cross the filter. They are low-pass, high-pass, band-pass, and band-stop. For analogical signals, the basic filter is achieved with various combinations of resistors, capacitors, and sometimes inductors, these are named passive filters. Active filters besides using passive element, it also uses active element such as transistors or operational amplifiers to provide desired voltage gains or impedance

characteristics. For, digital signals, filters are designed by using frequency band-pass low or high combinations convolved to the original signal to be filtered[15].

5.1.1.1 Ideal Filter

Ideally, filter should have the characteristics as shown in Figure 2.3. In practice, such characteristics are not possible to be achieved and the attenuation of the signal after the critical frequency is either exponentially increased or decreased. It is not abruptly decreased or increased as shown in figure [15].

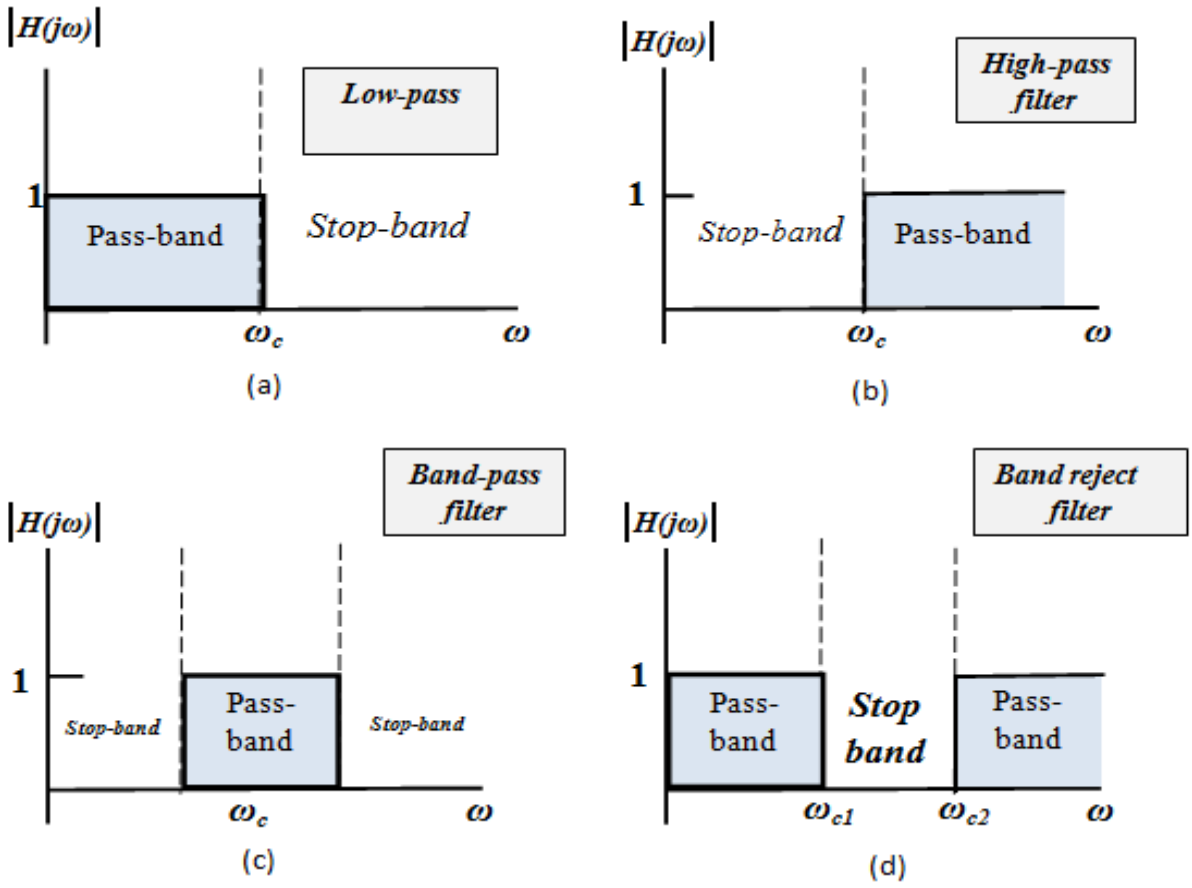


Figure 2.3:Ideal filter characteristic – (a) low-pass, (b)high-pass, (c) band-pass, (d)and band-stop.

Some famous digital filters are described in the following subsections.

5.1.1.2 Butterworth Filter

A low-pass Butterworth filter is an all-pole filter with a squared magnitude response given by:

$$|H_a(j\Omega)|^2 = \frac{1}{1 + (j\Omega/j\Omega_c)^{2N}} \tag{2.1}$$

The parameter *N* is the order of the filter (number of poles in the system function), and $\Omega_c$is the 3-dB cutofffrequency. The magnitude of the frequency response may also be written as:

$$|H_a(j\Omega)|^2 = \frac{1}{1 + \epsilon^2 \left(\frac{j\Omega}{j\Omega_p}\right)^{2N}} \tag{2.2}$$

Where:

$$\epsilon = \left(\frac{\Omega_p}{\Omega_c}\right)^N \tag{2.3}$$

The frequency response of the Butterworth filter decreases monotonically with increasing of the frequency, and as the filter order increases, the transition band becomes narrower. These properties are illustrated in Figure 2.4, which shows$|H_a(j\Omega)|$for Butterworth filters of orders $N = 2,4, 8,$ and 12. Because:[16]

$$|H_a(j\Omega)|^2 = H_a(s)H_a(-s)|_{s=j\Omega} \tag{2.4}$$

From the magnitude-squared function, we may write:

$$* G_a(s) = H_a(s)H_a(-s) = \frac{1}{1 + \left(\frac{s}{j\Omega_c}\right)^{2N}} \tag{2.5}$$

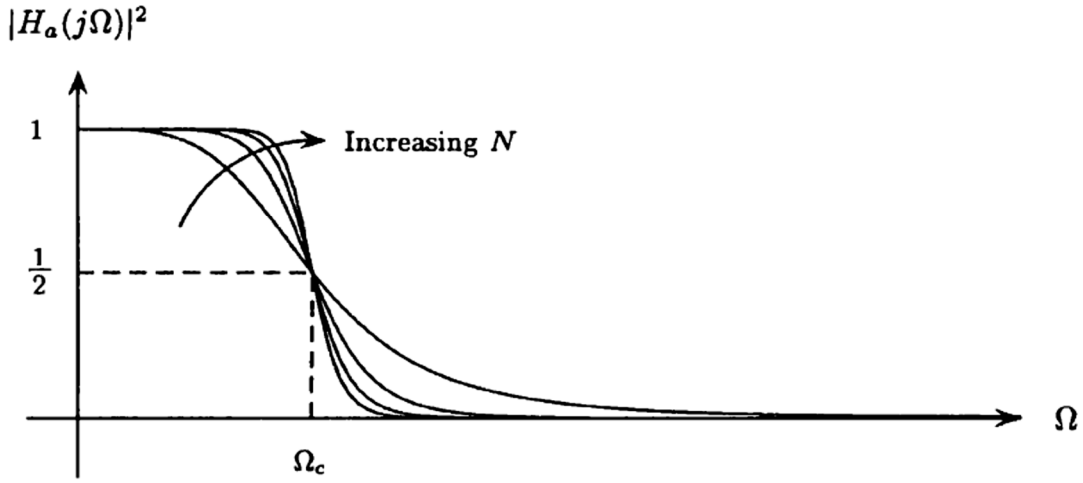


Figure 2.4: The magnitude of the frequency response for Butterworth filters of orders N= 2, 4, 8[16].

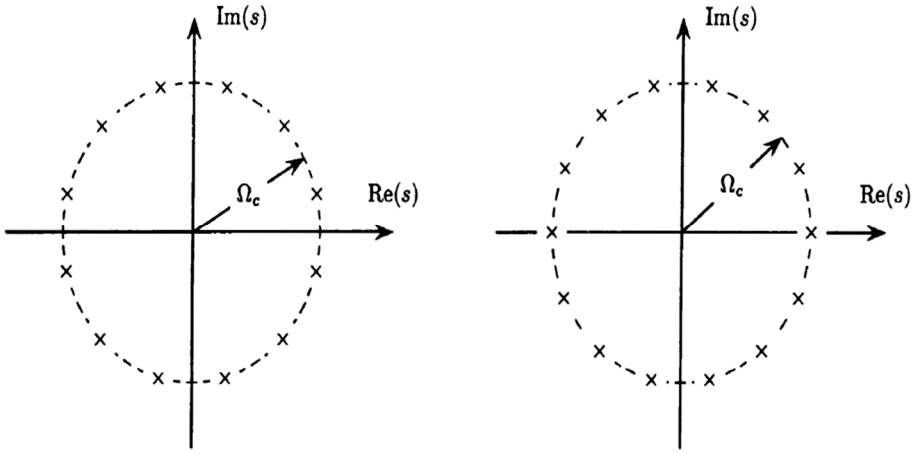
Therefore, the poles of $G_a(s)$are located at $2N$ equally spaced points around a circle of radiusΩ_c,

$$s_k = (-1)^{1/2N}(j\Omega) = \Omega_c \, exp\left\{j\frac{(N+1+2k)\pi}{2N}\right\} \qquad k = 0, 1, \dots, 2N-1 \qquad (2.6)$$

and are symmetrically located about the $j\Omega$-axis. Figure 2.4 shows these pole positions for $N = 6$ and $N = 7$.The system function,$H_a(s)$, is then formed from the N roots of$H_a(s)H_a(-s)$that lie in the left-half s-plane.

For a normalized Butterworth filter with $\Omega_c = 1$, the system function has the form[16]

$$H_a(s) = \frac{1}{A_N(s)} = \frac{1}{s^N + a_1 s^{N-1} + \cdots + a_{N-1}s + a_N} \qquad (2.7)$$



(a) Order N=6 (b) order N=7

Figure 2.5: The poles of $H_a(s)H_a(-s)$for a Butterworth filter of order N = 6 and N = 7 [16].

Table 2.1 lists the coefficients of $A_N(s)$for $1 \leq N \leq 8$. Given Ω_p, Ω_s, $\delta_p$and δ_s the steps involved in designinga Butterworth filter are as follows:

1. Find the values for the selectivity factor, k, and the discrimination factor, d. from the filter specifications.
2. Determine the order of the filter required to meet the specifications using the design formula: $N \geq \frac{\log d}{\log k}$ (2.8)

Table 2.1: The Coefficients in the System Function of a Normalized Butterworth Filter ($\Omega_c = 1$) For order $1 \leq N \leq 8$ [16].

N	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
1	1.0000							
2	1.4142	1.0000						
3	2.0000	2.0000	1.0000					
4	2.6131	3.4142	2.6131	1.0000				
5	3.2361	5.2361	5.2361	3.2361	1.0000			
6	3.8637	7.4641	9.1416	7.4641	3.8637	1.0000		
7	4.4940	10.0978	14.5918	14.5918	10.0978	4.4940	1.0000	
8	5.1258	13.1371	21.8462	25.6884	21.8462	13.1372	5.1258	1.0000

3. Set the 3-dB cutoff frequency, Ω_c to any value in the range [16]

$$\Omega_p \left[\left(1-\delta_p\right)^{-2}-1\right]^{-1/2N} \leq \Omega_c \leq \Omega_s[\delta_s^{-2}]^{-\frac{1}{2N}} \tag{2.9}$$

4. Synthesize the system function of the Butterworth filter from the poles of

$$G_a(s) = H_a(s)H_a(-s) = \frac{1}{1+\left(\frac{s}{j\Omega_c}\right)^{2N}} \tag{2.10}$$

that lie in the left-half s-plane. Thus,[16]

$$H_a(s) = \prod_{k=0}^{N-1} \frac{-s_k}{s-s_k} \tag{2.11}$$

Where:

$$s_k = \Omega_c exp\left\{j\frac{(N+1+2k)\pi}{2N}\right\} \qquad k = 0, 1, \dots, N-1 \tag{2.12}$$

5.1.1.3 Lattice Filters:

Lattice filters have a number of interesting and important properties that make them popular in a number of different applications. These properties include modularity, low sensitivity to parameter quantization effects, and a simple criterion for ensuring filter stability. In the following sections, we present the lattice filter structure for FIR filters, all-pole filters, and filters that have both poles and zeros [16].

FIR Lattice Filters

An FIR lattice filter is a cascade of two-port networks as shown in Figure 2.7. Each two-port network is defined by the value of its *reflection coefficient*,Γ_k.The two inputs, $f_{k-1}(n)$and$g_{k-1}(n)$are related to the outputs$f_k(n)$and $g_k(n)$by a pair of coupled difference equations [16]:

$$f_k(n) = f_{k-1}(n) + \Gamma_k g_k(n-1) \tag{2.13}$$

$$g_k(n) = g_{k-1}(n-1) + \Gamma_k f_{k-1}(n) \tag{2.14}$$

With the input to the first section being:

$$f_0(n) = g_0(n) + x(n) \tag{2.15}$$

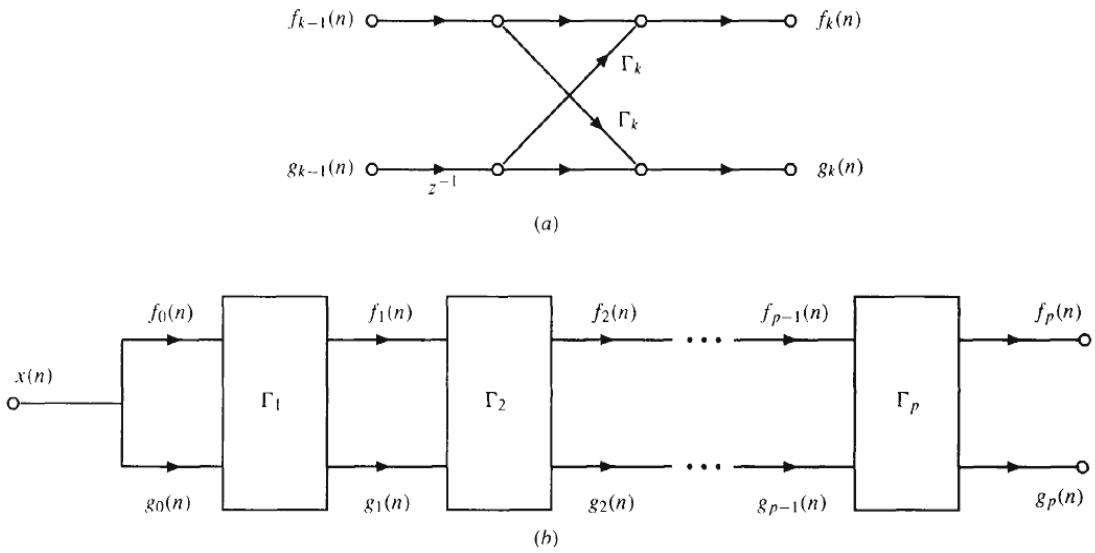


Figure 2.6:A*pth*-order FIR lattice filter. ***(a)*** The two-port network for each lattice filter module. ***(b)*** A cascadeof p lattice filter modules [16].

With $A_k(z)$the system function relating the input $x(n)$to the intermediate output$F_k(z)$,

$$F_k(z) = A_k(z)X(z) \tag{2.16}$$

These difference equations may be solved by induction to yield the following recurrence formula for$A_k(z)$:

$$A_k(z) = A_{k-1}(z) + \Gamma_k z^{-k} A_{k-1}(z^{-1}) \tag{2.17}$$

This is called the step-up recursion. The recursion is initialized by setting$A_0(z) = 1$. This recurrence formulaalso defines a recurrence relation for the coefficients $a_k(i)$of $A_k(z)$, which is:

$$a_k(i) = a_{k-1}(i) + \Gamma_k a_k(k-1) \qquad i = 1, 2, \dots, k-1 \tag{2.18}$$

$$a_k(k) = \Gamma_k \tag{2.19}$$

A simple way to write this recursion is in terms of vectors as follows:

$$\begin{bmatrix} 1 \\ a_k(1) \\ \vdots \\ a_k(k-1) \\ a_k(k) \end{bmatrix} = \begin{bmatrix} 1 \\ a_{k-1}(1) \\ \vdots \\ a_{k-1}(k-1) \\ 0 \end{bmatrix} + \Gamma_k \begin{bmatrix} 0 \\ a_{k-1}(k-1) \\ \vdots \\ a_{k-1}(1) \\ 1 \end{bmatrix} \tag{2.20}$$

All-Pole Lattice Filters

The structure for an all-pole lattice filter is shown in Figure 2.6. As with the FIR lattice, a p th-order all-pole filter is a cascade of p stages, with each stage being a two-port network that is parameterized by its reflection coefficient Γ_k.The two inputs, $f_k(n)$and $g_{k-1}(n)$, are related to the two outputs $f_{k-1}(n)$and$g_k(n)$by a pair of coupled difference equations:

$$f_{k-1}(n) = f_k(n) + \Gamma_k g_{k-1}(n-1) \tag{2.21}$$

$$g_k(n) = g_{k-1}(n-1) + \Gamma_k f_k(n) \tag{2.22}$$

The system function relating the input $x(n)$to the output $y(n)$is:

$$H(z) = \frac{1}{A_p(z)} \tag{2.23}$$

Where $A_p(z)$is the polynomial that is generated by the recursion given in Eq. (2.17).In addition, note that the system function relating $x(n)$to $w(n)$is an all-pass filter with a system function $A_{ap}(z)$givenby:

$$H_{ap}(z) = \frac{z^{-p}A_p(z^{-1})}{A_p(z)} \tag{2.24}$$

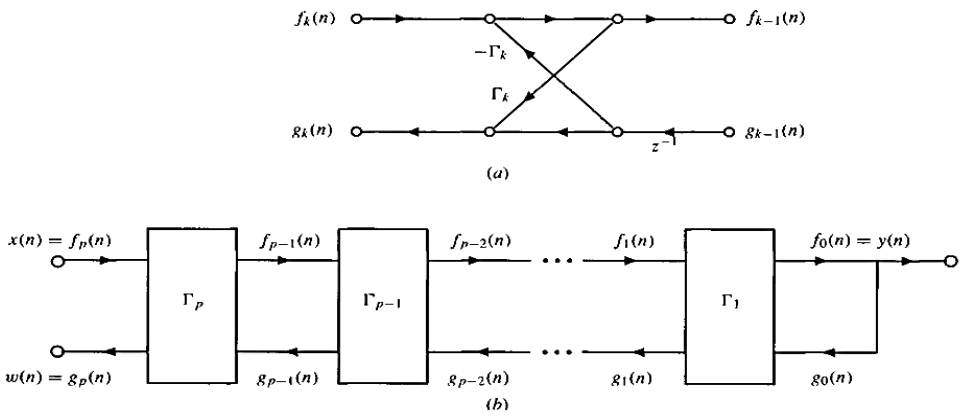


Figure 2.7: A pth-order all-pole lattice filter. (a) The two-port network for the kth stage of the all-pole lattice filter. (b) Cascade of p lattice stages[16].

IIR Lattice Filters

If $H(z)$ is an IIR filter with p poles and q zeros,

$$H(z) = \frac{B_q(z)}{A_p(z)} = \frac{\sum_{k=0}^{q} b_q(k)z^{-k}}{1 + \sum_{k=1}^{p} a_p(k)z^{-k}} \tag{2.25}$$

With$q \leq p$, a lattice filter implementation of $H(z)$ consists of two components. The first is an all-pole latticewith reflection coefficients $\Gamma_1, \Gamma_2, \ldots, \Gamma_p$, those implements$1/A_p(z)$. The second is a tapped delay line withcoefficients $c_q(k)$. The structure is illustrated in Figure 2.8 for the case in which $p = q$. The relationship betweenthe lattice filter coefficients $c_q(k)$and the direct form coefficients $b_q(k)$is given by [16]:

$$b_q(k) = \sum_{j=k}^{q} c_q(j)\, a_j(j-k) \qquad k = 0, 1, \ldots, p \tag{2.26}$$

Similarly, a recursion that generates the coefficients $c_q(k)$from the coefficients $b_q(k)$is:

$$c_q(k) = b_q(k) - \sum_{j=k+1}^{q} c_q(j)\, a_j(j-k) \qquad k = q-1, q-2, \ldots, 0 \tag{2.27}$$

This recursion is initialized with $c_q(q) = b_q(q)$.

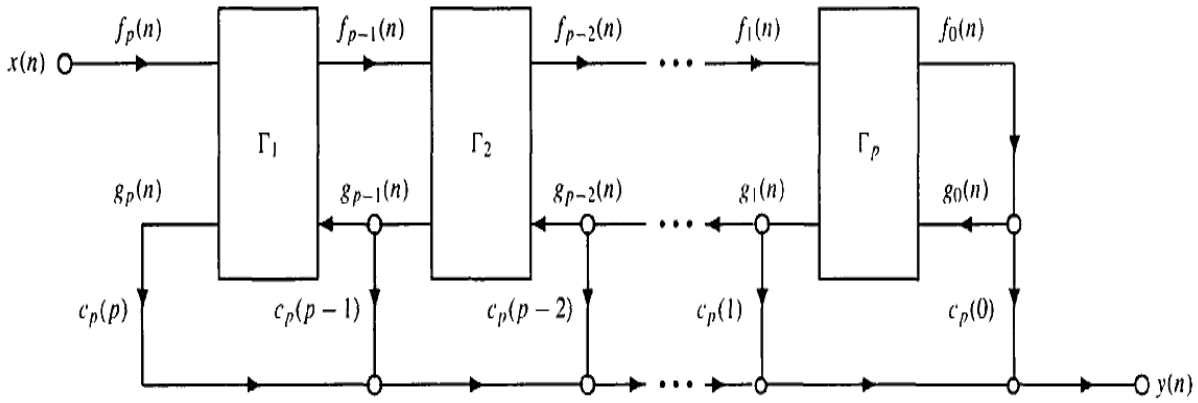


Figure 2.8: An IIR lattice filter with *p* poles and *p* zeros [16].

5.1.2 Signals Convolution

5.1.2.1 Signals Of The Same Dimensions

If the signals h and g are numerical and defined by the sequences $\{h\ (n)\}$ and $\{g\ (n)\}$ of dimension N, the convolution product is written:

$$s(n) = h(n) * g(n) = \sum_{K=0}^{N-1} h(k) g(n-k) \tag{2.28}$$

The signals h and g are both defined over the interval $[0, N-1]$. However in the operation of reversal of the signal g the index n - k will be likely to run of $[-N+1, N-1]$. Since g is only defined on the interval $[0, N-1]$ it is important toset the rules of calculation and these rules can be different according to the objective that one is fixed. We can see that if we start from the signal g returned, this signal will remain in coincidence with the signal h not returned as long as n will belong to the interval $[0, 2N-1]$. The dimension of the convoluted signal is therefore $2N-1$ if the starting signals are both a clear way to establish this property is to consider that the signal g exists only if the quantity $n-k$ belongs to the interval $[0, N-1]$ with k contained in the same interval [17].

Suppose that k is equal to either the index $k = 0$ or the index $k = N-1$. The signal $g(n-k)$ will exist if

$$\begin{cases} 0 \leq n \leq N-1 \\ 0 \leq n-N+1 \leq N-1 \end{cases} \Leftrightarrow 0 \leq n \leq 2N-2 \qquad (2.29)$$

which leads to a convoluted signal of dimension $2N-1$

To perform the numerical calculation we adopt the following rules [17]:

Rule n°1

We perform the operation considering that in the sum contained in the Equation 2.28 samples with an index outside the range $[0, N-1]$ will be zero. The problem obviously does not arise for h but clearly for g.

We have then:

$$s(n) = h(n) * g(n) = \sum_{K=0}^{N-1} h(k)g(n-k) \qquad n \in [0;\ 2N-2] \qquad (2.30)$$

Rule n°2

The operation is performed considering that the samples outside the interval$[0, N-1]$ have the same value as those contained in the interval which corresponds to periodize the returned signal.

Rule n°3

We do the calculation on an interval half of the dimension of the signal which comes back indirectly using rule 2.

It is clear that the proposed rules for this calculation do not all lead to the same result. This will be discussed later.

The following example shows how the convolution product is performed in the case of digital signals. We will consider the following signals:

$$\begin{matrix} n & 0 & 1 & 2 \\ h(n) & 1 & 2 & 3 \end{matrix} \quad / \dim h = 3 \tag{2.31}$$

and

$$\begin{matrix} n & 0 & 1 & 2 \\ g(n) & 1 & 1 & 2 \end{matrix} \quad / \dim g = 3 \tag{2.32}$$

The graph of these signals looks like this:

To carry out the convolution operation, it is now necessary to return the signal g starting with$n = 0$. It is trivial to note that $g(-k)$ which is nothing else that $g(n - k)$for $n = 0$.

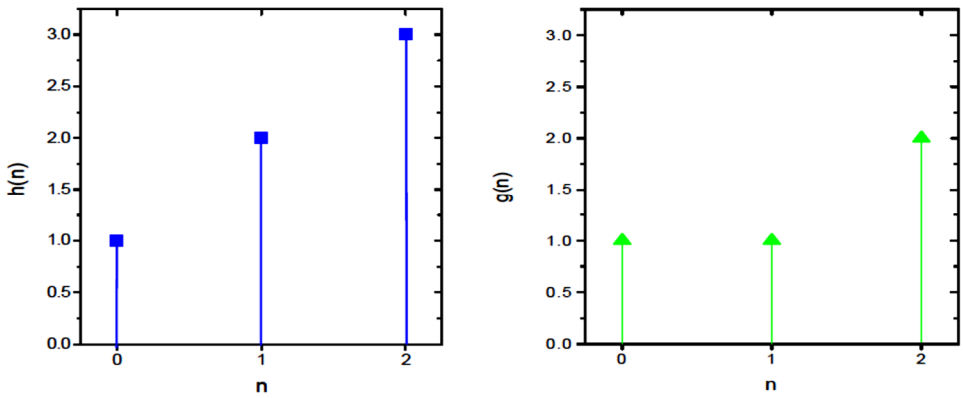


Figure 2.9: Graph of the tow signals [17].

During this operation we obtain the symmetry of g by relative to the y-axis. Indeed to check it just as k $K = -k$ and it is immediately see that [17]:

$$\begin{matrix} -k & 0 & -1 & -2 \\ g(n) & 1 & 1 & 2 \end{matrix} \tag{2.33}$$

If we fix the rule n◦1 calculation we see that we will not keep in the signal returned only $g\ (0)$ to perform the summation contained in the convolution product.

The first sample of the convoluted signal

$$s(0) = \sum_{k=0}^{2} h(k)g(-k) = h(0)g(0) = 1 \tag{2.34}$$

For $n\ =\ 1$ the signals $h\ (k)$ and $g\ (1\ -\ k)$ have a common intersection in $k\ =\ 0$ and

$k\ =\ 1$ and the second sample of the convoluted signal becomes:

$$s(1) = \sum_{k=0}^{2} h(k)g(1-k) = h(0)g(0) + h(1)g(0) = 1 + 2 = 3 \tag{2.35}$$

By gradually sliding the signal $g\ (n\ -\ k)$ from $n\ =\ 0$ to $n\ =\ 2N\ -\ 2$ onthe axis of the abscissa is obtained the product of convolution of the two digital signals.

The samples contained in the convoluted signal are:

$$\begin{matrix} n & 0 & 1 & 2 & 3 & 4 \\ s(n) & 1 & 3 & 7 & & \end{matrix} \tag{2.36}$$

5.1.2.2 Signals of Different Dimensions

Now, we approach the case of two signals h and g of different size.

We put $dimh\ =\ N$ and $dimg\ =\ M$. By definition we always have [17]:

$$s(n) = h(n) * g(n)s(0) = \sum_{k=0}^{N-1} h(k)g(n-k) \tag{2.37}$$

In this case $n - k$ must belong to the interval $[0, M\ -\ 1]$ and k to the interval $[0, N\ -\ 1]$.

Therefore, the index n must verify:

$$\begin{cases} 0 \le n \le M-1 \\ 0 \le n-N+1 \le M-1 \end{cases} \Leftrightarrow 0 \le n \le N+M-2 \tag{2.38}$$

which shows that the dimension of the convoluted signal is $N + M - 1$.

In this case we use the calculation rule n◦1 [17].

5.2 Spike Detection and Alignment

5.2.1 Spike Detection

Spike detection as the first basic step, it is very important in data analyzing and

classification, since the quality of resulting data depends crucially on a particular detection technique used among them the threshold application, and there are another class of spike-detection algorithms are based on detecting changes in the energy of the signal. Some of these algorithms are the nonlinear energy operator (NEO) and the Teager energy operator (TEO).

5.2.1.1 Teager Energy Operator (Teo)

The TE operator of a complex-valued signal $x(t)$ isdefined as

$$\Psi_C[x(t)] = \dot{x}(t)\dot{x}^*(t) - \frac{1}{2}[\ddot{x}(t)x^*(t) + x(t)\ddot{x}^*(t)] \quad (2.39)$$

When $x(t)$ is real, (2.39) reduces to the TE of a real-valued signal, which is defined as

$$\Psi_R[x(t)] = \dot{x}^2(t) - x(t)\ddot{x}(t) \quad (2.40)$$

Furthermore, writing the complex signal $x(t)$ as a function of itsreal and imaginary parts:

$$x(t) = x_r(t) + jx_i(t) \quad (2.41)$$

then applying the complex TE operator of (2.39), we obtain:

$$\Psi_C[x(t)] = \Psi_C[x_r(t) + jx_i(t)] = \dot{x}_r^2(t) + \dot{x}_i^2(t) - x_r(t)\ddot{x}_r(t) - x_i(t)\ddot{x}(t) \quad (2.42)$$

Hence, the TE of a complex signal is equal to the sum of the Teager energies of its real and imaginary parts:

$$\Psi_C[x(t)] = \Psi_R[x_r(t)] + \Psi_R[x_i(t)] \quad (2.43)$$

It is proposed, the following TE operator for complex-valued signals:

$$\Psi_C[x(t)] = \|\dot{x}(t)\|^2 - Re|x^*(t)\ddot{x}(t)| \tag{2.44}$$

Substituting the complex-valued signal $x(t)$ by its real and imaginary parts in (2.44), it can easily be show that (2.43) holds also for Maragos and Bovik's definition, although ours exhibits the symmetry of the operator more clearly. Note that both definitions yield a real quantity, as expected for an energy operator [18].

5.2.1.2 Nonlinear Energy Operator (Neo)

The Non-linear Energy Operator (a.k.a. Tieger operator) is known to be very effective in spike detection, especially for signals with very low SNR (Signal to Noise Ratio). The NEO output is proportional to the square of the instantaneous product of a signals amplitude and frequency [19], described by

$$v_{(x(t))} = (\dot{x}(t))^2 - x(t) \cdot \left(\ddot{x}(t)\right) \tag{2.45}$$

It can easily be shown that the output of the NEO is proportional to the product of the amplitude and frequency of the input signal ……..[20].

For a discrete-time sequence $x(t)$, the NEO is given as:

$$\Psi[x(n)] = x^2(n) - x(n+1)x(n-1) \tag{2.46}$$

The NEO has been used for the amplitude and frequency demodulation, the analysis of speech signals, and recently for the detection of a spiky waveform in (ECoG) all of these studies have made the use of the fact that the NEO can simultaneously consider the "instantaneous" amplitude and frequency information of the input signal. By the "instantaneous" amplitude and frequency, we mean the amplitude and frequency of the dominant sinusoidal component at any particular time. More formally, they can be defined by the Hilbert transform pair. When an action potential is fired, it is possible to see an instantaneous increment of signal amplitude and frequency using time-frequency analysis such as short-time Fourier transform or Wigner–Ville distribution. The output of the NEO is convolved with a Bartlett window in order to eliminate the spurious peaks due to the cross terms and background noise [20].

5.2.1.3 Threshold Application

Thresholding is an often-used method of spike detection for implantable neural signal processors due to its computational simplicity. A means for automatically selecting the threshold is desirable, especially for high channel count data acquisition systems. Estimating the noise level and setting the threshold to a multiple of this level is a computationally simple means of automatically selecting a threshold [21].

In any spike detection algorithm**,** the choice of threshold for detection of a spike event is a tradeoff between avoiding false positives (i.e., Type I errors) and false negatives (i.e., Type II errors). A low threshold will capture the most spike events but will erroneously admit many noise events. A high threshold rejects the most noise but also misses the most spikes. When a statistical model of noise is used, the threshold can be calibrated to a desired ratio of Type I to Type II errors. In most cases, a more permissive threshold is desirable because windows of noise can be removed in later stages of processing [22].

The threshold for NEO detection is a statistical calculation, which the mean of the NEO filtered out-put multiplied by a factor [23] The formula for NEO threshold is given in (2.47):

$$Thr = C \times \frac{1}{N} \sum_{n=1}^{N} \Psi[x(n)] \tag{2.47}$$

Where; N is the number of samples of the signal, and C is the multiplying factor found empirically. This factor was tested in [23] on several neural datasets and an approximate value of 8 gave the best detection results at multiple SNRs. Since the factor is a power of 2, it is relatively easily implemented in hardware, and thus this approximation is suitable when the threshold needs to be determined in hardware on-chip [24].

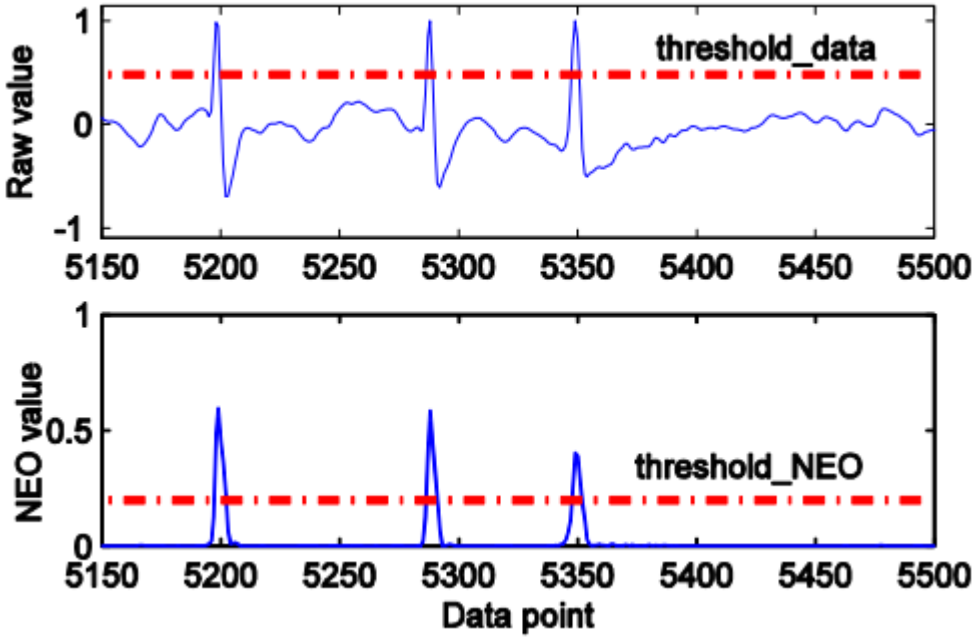


Figure 2.10:Spike detection using thresholding in NEO and raw data.

Figure 2.10illustrates how thresholds are applied to raw and NEO data. It is clear that NEO data provide a better contrast between spike and noise data. Outside the spikes, data are smoother. Therefore $x(n), x(n-1)$and $x(n+1)$are similar in magnitude, leading to $\Psi[x(n)]$close to zero. This is true even when raw data is large in magnitude but not spiky. NEO is less prone to error even when noise accumulates to a high voltage level. The choice of NEO threshold value depends on data set and is normally obtained after a training period. So, threshold value in NEO is defined as:[25]

$$Thres_{NEO} = 8\frac{1}{N}\sum_{1}^{N}\Psi[x(n)] \tag{2.48}$$

5.2.2 Alignment

When spike detection is performed in the digital domain, whenever the voltage signal crosses a threshold, a window is applied and a spike waveform is captured. At this point, each spike is essentially aligned to the point of the threshold crossing. However, sampling jitter combined with noise effects may leave features of interest, such as maximum and minimum values, misaligned. This temporal misalignment has the effect of increasing the spread of points in feature space, making clustering more difficult. Thus, alignment should be performed prior to classification.

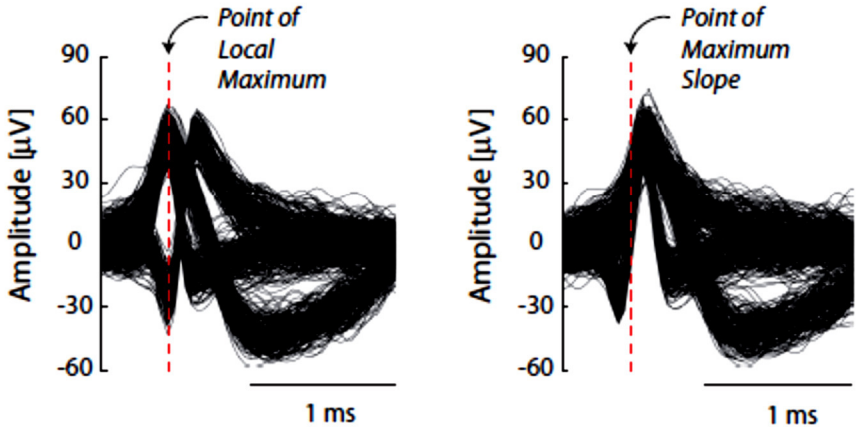


Figure 2.11: Examples of two different alignment methods. Left: alignment to maximum amplitude, Right: alignment to maximum slope.

The alignment process usually begins by up-sampling the signal (using an interpolation method such as cubic spline) to help reduce the effects of sampling jitter. Then, the signal is aligned to some event in time. The aligned spikes may be down-sampled to the original sampling rate after alignment.

The most common method of temporal alignment is to align each spike to the point of its maximum amplitude (Fig. 2.11) .Alignment to the point of maximum slope (Fig. 2.11) has also been proposed, which is intuitive since the rising slope of the action potential has biological significance. This method would be especially convenient if discrete derivatives were already being used for feature extraction. Others have proposed alignment to the maximum of an energy measure such as the NEO, which would be convenient if NEO were already being used for spike detection. Similarly, alignment to the maximum integral would be convenient if the integral transform were being used for feature extraction. Indeed, it would be convenient to perform alignment with respect to any measure that is already being calculated in the sorting process.

Although the aforementioned alignment methods will usually improve classification accuracy, alignment to a metric that is derived from the whole spike rather than from a single point may be less susceptible to the effects of background noise. One example of such a metric is the spike's center of mass. Note that all of the algorithms that have been described in this section are completely automatic and real-time[11]

5.3 Feature Extraction

The implementation of a generalized feature evaluation step in the spike sorting process does not only provide the opportunity to use a mixture of PCA and Wavelet features for the classification step, but also allows the inclusion of other features that are, for example, based on the shape of the measured spikes themselves.Certain characteristics of the spike, e.g. its negative or positive amplitude, as well as various gradients or its signal energy can be used as additional candidates. On the one hand, such geometry based features are usually vulnerable to noise signals and voltage offsets. On the other hand, they can be very potent in particular situations, in which such a feature can specifically highlight a difference between two spike shapes. Furthermore, these features can be quickly and easily calculated and represent no imminent threat to the algorithm's computational complexity [12].

5.3.1 Geometric Features

The calculation of the positive and negative amplitude is basically self-explanatory. The left and right spike angles represent the left and the right gradient of the characteristic spike minimum to the point in the signal, at which it crosses the margin of 50% of that value. The choice of this particular margin can be explained by low liability to the noise components of the signal. The resulting value can easily be transformed into an angular measure, using basic trigonometric functions. In addition, the spike duration can be approximated as the distance between the intersection points of both gradients with the $0\ V$ level. The positive and negative signal energy of a continuous-time signal $s(t)$ canbe approximated based on the following equation:

$$Es = \int_{-\infty}^{\infty} s^2(t)dt \tag{2.50}$$

Likewise, the signal energy can also be calculated for discrete-time signals by summing the squared discrete signal values. In this case, the positive and negative energy components of the spike are calculated in particular to separate mono-polar from bipolar spike shapes more accurately [12].

5.3.2 Principal Component Analysis:

The strategy behind PCA is to apply a transformation in the spikes. The difference is that the new base in which the spikes are described is composed of a set of orthogonal vectors, which eliminates the redundancy in its representation. In turn, the data reduction is achieved by considering only the coordinates that increase the capability of distinguishing the neurons that elicited the spikes. According to the literature, a minimum set of 3 coordinates already allows capturing more than 90% of the variance in the spike data. The mathematical definition is presented by the next expression, where each spike is expressed as a series of Principal Component Coefficients (PCC) c_i, N is the number of samples in a spike s, and PCi is the $i\,th$ PCC:

$$c_i = \sum_{n=1}^{n} PC_{i,n} \cdot s_n \tag{2.51}$$

These PCCs are obtained by the eigen value decomposition of the covariance matrix of the data. The disadvantage of this algorithm is that it is oriented by the maximum variance in the data, which is not necessarily the best set of features to identify which neuron the spike corresponds to. Nonetheless, this is by far the most used algorithm for the FE stage and it is presented here for comparison purposes since it cannot be used to implement a real-time solution [14].

5.4 Spike Classification and Clustering

The classification of the detected spikes according to the generator neuron is accomplished in this last stage. It analyzes these spikes (already represented through the features generated in the previous stage, FE) and classifies them based on their similarity. There is no usual sub-division of this stage in the literature. In the first approaches, this was done manually and the quality of the results depended on the skills of the person in charge of the analysis. Since then, many other methods have been proposed to automate the clustering step. One of the main differences among these automatic methods is the capability to run in real-time. Several methods provide good accuracy and are robust to SNR differences but need to process the whole recorded data and it is not possible to apply them in a real-time application. This is the case of the K-means and Super-paramagnetic Clustering (SPC) algorithms. On the other hand, several real-time approaches have been proposed, which process and adapt the sorting process in real-time (e.g., OSort). Although such approaches are not as accurate as their offline counterparts, they can be integrated and used for the

development of neural-prosthesis. Alternatively, other approaches operate in real-time, but require a training phase in order to adjust their parameters to the specificity of the circumstances. However, one of the main drawbacks of such approaches is the inability of the system to detect spikes from neurons that did not fire during the training phase [14].

5.4.1 K-Means

K-means has become a standard in clustering applications because of its simplicity and accuracy. This is a simple approach, also called "nearest neighbor". It was proposed by Hartigan 1975 and a more efficient version by the same author was put forward in 1979. The main idea is to partition the P points in D dimensions into K clusters, minimizing the sum of the squares in each cluster. With a large dataset, and when $K > 2$, it is impractical to require a global minimum when the solution has the minimum sum of squares in all clusters, since a local minimum in one cluster negatively affects another cluster. Instead, a local optimum is expected, where no movement of points from one cluster to another will reduce the sum of squares in the cluster. In k-means, the average of each group is called a centroid. If the centroid must be on a real data point, and not on the real average, it is called k-medoids clustering. To calculate the similarity, a distance metric form must be used. We used Euclidean distance in our implementation, but other measures such as the distance from Manhattan might make more sense for other problems. K-means is a fast and well known algorithm. A high-level overview is shown in the following algorithm[14,26].

Algorithm High-level serial k-means [26]

1: Distribute K centroids according to some heuristic, as explained later inthis section.
2: **repeat**
3: **for all** point in points **do**
4: calculate which centroid is nearest, and assign point to that centroid
5:**end for**
6:**for all** centroid in centroids **do**
7: recalculate centroid position as a mean of all the current points inits cluster
8:**end for**
9: **until** stop criteria (number of iterations, number of membership changes,distance moved, etc)

5.4.2 Bayesian Clustering

In BC, the data is seen as having a statistical distribution for which a model is wanted, i.e. the clustering. One of the main advantages is the quantification of the certainty of each spike generated by each neuron (cluster) resorting to the Bayes' rule:

$$p(c_k|x,\theta_{1:K}) = \frac{p(x|c_k,\theta_k)p(c_k)}{\sum_k p(x|c_k,\theta_k)p(c_k)} \tag{2.52}$$

where$c_k$is a cluster, x is the spike data and $\theta$is used to represent the parameters of each class, e.g.$\theta_{1:k} = \mu_1, \sum_1, \cdots, \mu_k, \sum_k$.Then, the class parameters are optimized by maximizing the likelihood of the data:

$$p(x_{1:N}|\theta_{1:K}) = \prod_{n=1}^{N} p(x_n|c_k,\theta_{1:K}) \tag{2.53}$$

An additional step can be performed in order to increase the robustness to outliers: the creation of a 'background' cluster with a low weight. It is responsible for accommodating such undesired issues and, at the end, through the distribution of probabilities of each cluster, it is possible to see how well separated the actual clusters are. The possibility to calculate the likelihood helps the experimenter to make decisions about the isolation of the spikes [14].

5.4.3 Super-Paramagnetic Clustering

Super-paramagnetic clustering (SPC) is a non-parametric classification algorithm presented by Blatt et al. SPC is based on the physical properties of an inhomogeneous ferromagnetic model. And on the Potts model theory, which describes the interaction between spins on a crystal lattice, and is used to describe the behavior of ferromagnets at varying temperatures[26].

This algorithm proposes an automatic classification of points without resorting to hypotheses and has been successfully applied to the sorting of points by Quiroga et al.

A spin is simply a point on the lattice with a state "spin "q, ranging from 1 to Q. The points on the lattice are called spins because of the way they will rotate to align with a magnetic field The increase in temperature will increase the entropy, which reduces the influence of neighboring spins, in relation to the Curie temperatures (Tc) of the ferromagnetic, where the ferromagnetic becomes paramagnetic. Over this temperature, the

alignment will be random, and a ferromagnetic will no longer be influenced by a magnetic field (the magnetic susceptibility tends to 0). For example, the Curie temperature of iron (Fe) is 770◦C, and above this temperature, the iron will no longer attract a magnet[26].

Blatt et al. give a good description of the process, in an example with three dense regions: At high temperatures, the system is in a disordered (paramagnetic) phase. When the temperature is lowered, a transition to a super-paramagnetic phase occurs; rotates in the same high density region become completely aligned, while different regions unordered. As the temperature is further lowered, the efficiency coupling between the three clusters (induced by diluted background spins) increases until it aligns. [...] we call this "phase" of ferromagnetic aligned clusters.

Therefore, the algorithm has two main steps: locate the super-paramagnetic phases and determine how the points cluster in these phases[14,26]

Superparamagnetic Clustering algorithm [11]

---

1. Calculate the Euclidean distance matrix $D = (d_{ij})$.

2. Identify the K nearest neighbors of each point v_i. $v_i$and$v_j$are considered neighbors if $v_i$is one of the K nearest neighbors of v_j and v_j is one of the K nearest neighbors of vi.

3. Assign a Potts spin $si = 1, 2, \dots, q$ to each point v_i randomly (or set all $s_i = 1$), where$q$ is a constant representing the number of possible spins. Note: The value chosen for $q$does not imply anything about the number of clusters.

4. Calculate the interaction strength J_{ij} between neighboring points $v_i$and v_j , where

$$J_{iJ} = \begin{cases} \frac{1}{k} exp\left(-\frac{d_{ij}^2}{2a^2}\right) & if\ v_i\ and\ v_j\ are\ nieghbors, \\ 0 & other\ wise, \end{cases} \tag{2.54}$$

and a is the average of all d_{ij} 's between neighboring points.

5. For each temperature (e.g. $T = 0 : 0{:}02 : 0{:}2$), perform the following Monte Carlo simulation of iterations $m = 1 : M$:

(a) Assign a frozen bond between nearest-neighbor points v_i and v_j with probability:

$$p_{ij}^f = 1 - exp\left(-\frac{J_{ij}}{T} \cdot \delta s_i s_j\right), \qquad where\ \delta_{s_i,s_j} = \begin{cases} 1\ if\ s_i = s_j, \\ 0\ \text{otherwise}, \end{cases} \tag{2.55}$$

(b) Generate a random number x from a uniform distribution on $[0, 1]$. If $x < p_{ij}^{f}$there is a bond between $v_i$and v_j.

(c) Define clusters as all points that are connected by a bond.

(d) Definec^m, where

$$C_{ij}^{m} = \begin{cases} 1 & if\ v_i\ and\ v_j\ are\ nieghbors, \\ 0\ \text{otherwise}, \end{cases} \tag{2.56}$$

6. Calculate the two-point connectedness C_{ij} , where

$$C_{ij} = \frac{1}{M}\sum_{m=1}^{M} C_{ij}^{m} \tag{2.57}$$

7. Calculate the spin-spin correlation function:

$$G_{ij} = \frac{(q-1)C_{ij}+1}{q} \tag{2.58}$$

8. If $Gij > \theta$ where θ is a pre-defined threshold, v_i and v_j belong to the same cluster.

9. Assign cluster labels to observations according to G.

5.4.4 OSort

OSort is a MATLAB implementation of a model-based unsupervised online spike sorting algorithm. OSort has already been designed focusing on the application in real time. It was first introduced by Rutishauser One of its important features is that it removes the FE phase because it directly calculates the distance between an income peak and the clusters Estimating the number of neurons present, as well as the assignment of each peak to a neuron, is based on a distance metric between two peaks (Rutishauser, 2006). Based on this distance, a threshold is used to decide the number of neurons present and to assign each peak only to a group of neurons, or to a noise group if it is not bearable. The threshold is calculated from the noise properties of the signal and is equal to the mean square deviation of the signal, calculated with a sliding window. OSort's main advantage over its competitors is that it can be used online, allowing for real-time sorting during an experiment [27,14].

OSort clustering algorithm [11]

1. Initialization: Assign the first data point to its own cluster.

2. Calculate the Euclidean distance between the next data point and each cluster centroid.

3. If the smallest distance is less than the merging threshold TM, assign the point to the nearest cluster and recompute that cluster's mean using the N most recent points. Otherwise, start a new cluster.

4. Check the distances between each cluster and every other cluster. If any distanceis below the sorting threshold TS, merge those two clusters and recompute its mean. Steps 2 and 4 are then repeated indefinitely. In the simplified version of this algorithm,T= TS, which is equal to the variance of the data computed continuously on a long (1 minute) sliding window. Note that when computing cluster centroids, only the N most recent points are used. This helps to account for electrode drift, since the clusters are allowed to drift as well.

5.4.5 Template Matching

Another important technique is TM, which is based on a comparison between the input signal under analysis and a predefined template. This template is obtained from an arbitrary training set and aims to describe the average of the main features. After the comparison, it is decided if the input signal is considered as a new spike or not. Three different approaches are considered: Matched-Filter (MF), NEO combined to MF and the absolute value (AV) combined to MF, which main difference regards the pre-emphasis step. Hence, for a pure MF approach,[14]

1. a convolution is applied directly to the raw signal with the predefined template;
2. a convolution is applied to the pre-emphasized signal by the AV algorithm when the MF is allied with AV;
3. Finally, a convolution is applied to the pre-emphasized signal by the NEO algorithm when the MF approach is combined with the NEO algorithm.

In other words, the First spike will be ranked as first form or a 'template', so the following spike will be compared to the previous one:

- If the next spike matches with the first one, it will be recorded as a new spike in the same class.
- If the next spike does not match with the first one, a new form is detected and ranked in a new class.

Matching calculation:

We calculate the quadratic error between the template and the new spike using the formula bellow:

$$err = \sum_{i=1}^{n} \big(temp(i) - x(i)\big)^2 \tag{2.59}$$

To decide or compare this error to the threshold:

- If $err_q < threshold$, so there is a resemblance.
- If not, there is no resemblance, so it is a new form.

So as a result, if there is resemblance, the ranked form may change a little because of having the median form with the template, or applying a short algorithm on this form:

Template matching algorithm

New spike detected

$\text{erq} = \sum_{i=1}^{n} \big(temp(i) - x(i)\big)^2$

if $\text{erq} < seuil \quad (resemblance)$

$\text{totspikes} = \text{totspikes} + 1$

$\text{spike(ti)} = 1$

$\text{forme} = \text{forme} * (\text{totspikes} - 1) + \text{new spike}$

else

$\text{newforme} = \text{newspike}$

$\text{spike(ti)} = 1$

$\text{totspike} = 1$

end

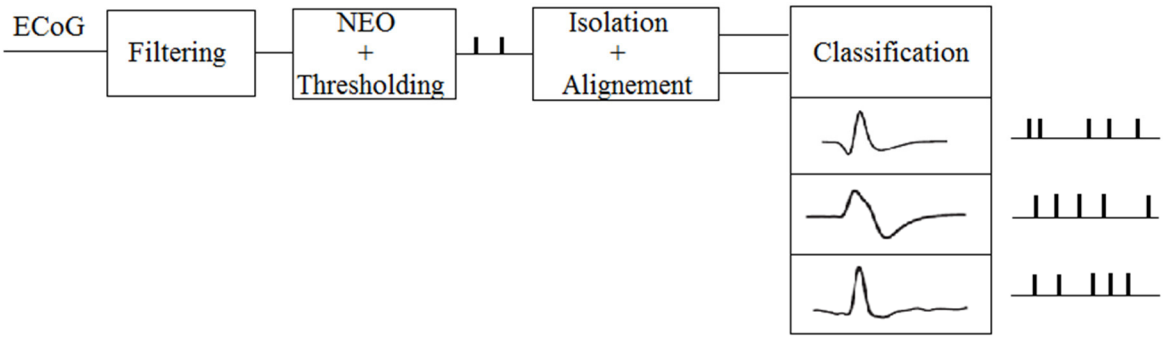


Figure 2.12: Final schema of the online spike sorting.

6 Conclusion

Spike detection, classification and sorting become more important for any brain signal uses such as brain computer interfaces and medical analyzing of brain behavior study.

In this chapter, we tried to present the four basic and principal levels in spike sorting process: filtering, detection, feature extraction and finally, spike classification and clustering. Also, we talked about the new technique that is Online Spike Sorting, and the different wide known algorithms used in spikes classification and clustering.

In the following chapter, we will present the results of an online spike sorting on a real data ECoG signal.

CHAPTER 3 : RESULTS REPRESENTATION

1 Introduction

In this chapter, we'll present results of the different spike sorting steps applied on real data of a rat pressing a lever to bring water.

The application of spikes sorting is done using MATLAB software where it is possible to program our own function with a simple instruction set. At first we will give a brief presentation of the MATLAB software then we move to the data collection part and finally the representation of results.

2 MATLAB Software

MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include:

- Math and computation
- Algorithm development
- Data acquisition
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including graphical user interface building

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows users to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of time it would take to write a program in a scalar non interactive language such as C and FORTRAN.

The name MATLAB stands for ***Matrix Laboratory***. MATLAB was originally written to provide an easy access to matrix software developed by the LINPACK and EISPACK projects. Today, MATLAB engines incorporate the LAPACK and BLAS libraries, embedding the state of the art in software for matrix computation.

MATLAB has evolved over a period of years with inputs from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in

mathematics, engineering and science. In industry, MATLAB is the tool of choice for high-productivity research, development, and analysis.

MATLAB features a family of add-on application-specific solutions called ***toolboxes***. Very important to most users of MATLAB, toolboxes allow users to ***learn*** and ***apply*** specialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others.

2.1 The MATLAB System

The MATLAB system consists of five main parts:

2.1.1 Development Environment:

This is the set of tools and facilities that help using MATLAB functions and files. Many of these tools are graphical user interfaces. It includes the MATLAB desktop and Command Window, a Command History, an editor and debugger, and browsers for viewing help, the workspace, files, and the search path.

2.1.2 The Matlab Mathematical Function Library:

This is a vast collection of computational algorithms ranging from elementary functions, like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix Eigen values, Bessel functions, and fast Fourier transforms … etc.

2.1.3 The MATLAB Language:

This is a high-level matrix/array language with control of flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and impropriate throw-away programs, and "programming in the large" to create large and complex application programs.

2.1.4 Graphics:

MATLAB has extensive facilities for displaying vectors and matrices as graphs, as well as annotating and printing these graphs. It includes high-level functions for two

dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level functions that allow you to fully customize the appearance of graphics as well as to build complete graphical user interfaces on your MATLAB applications.

2.1.5 The MATLAB External Interfaces/API:

This is a library that allows users to write C and FORTRAN programs that interact with MATLAB. It includes facilities for calling routines from MATLAB(dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files.

2.2 MATLAB Simulink

Simulink is an environment for simulation and model-based design for dynamic and embedded systems. It provides an interactive graphical environment and a customizable set of block libraries that let you design, simulate, implement, and test a variety of time-varying systems, including communications, controls, signal processing, video processing, and image processing.
Simulink offers:

- A quick way to develop models in contrast to text based-programming language such as C.
- Simulink has integrated solvers. In text based-programming language such as C you need to write your own solver.

3 Experimentation and Data Representation

3.1 Animal Training and Behavioral Tasks

The study, approved by the Institutional Animal Careand Use Committee at the National Chiao Tung University, was conducted according to the standards established in the Guide for the Care and Use of Laboratory Animals. Four male Wistar rats weighing 250-300g (BioLASCO Taiwan Corp., Ltd.) were individually housed on a 12 h light/dark cycle, with access to food and water *ad libitum*. Dataset was collected from the motor cortex of awaked animal performing a simple reward task. In this task, male rats (BioLACO Taiwan Co.,Ltd) were trained to press a lever to initiate a trial in return for a water reward. The animals were

water restricted 8-hours/day during training and recording session but food were always provided to the animal *ad lib* every day.

3.2 Chronic Animal Preparation And Neural Ensemble Recording

The animals were anesthetized with pentobarbital (50mg/kg i.p.) and placed on a standard stereotaxic apparatus (Model 9000, David Kopf, USA). The dura was retracted carefully before the electrode array was implanted. The pairs of 8 micro-wire electrode array (no.15140/13848, 50m in diameter; California Fine WireCo., USA) are implanted into the layer V of the primary motor cortex (M1). The area related to forelimb movement is located anterior 2-4 mm and lateral 2-4 mm to Bregma. After implantation, the exposed brain should be sealed with dental acrylic and a recovery time of a week is needed.

During the recording sessions, the animal was free to move within the behavior task box (30 cm×30 cm× 60cm), where rats only pressed the lever via the right forelimb for receiving 1-ml water reward as shown in figure 3.1. A Multi-Channel Acquisition Processor (MAP, Plexon Inc., USA) was used to record neural signals. The recorded neural signals were transmitted from the head stage to an amplifier, through a band-pass filter(300Hz-3kHz),and sampled at 20 kHz per channel. Simultaneously, the animal's behavior was recorded by the video tracking system (CinePlex, PlexonInc.,USA) and examined to ensure that it was consistent for all trials included in a given analysis. The obtained data; after spike sorting process; was composed of 48 channels (number of neurons) containing succession of '1' separated by long silence of '0'. Another representation is used based onthe rate of spike smoothed with a Gaussian window. In next parts, we'll show results of these explained steps with their figures.

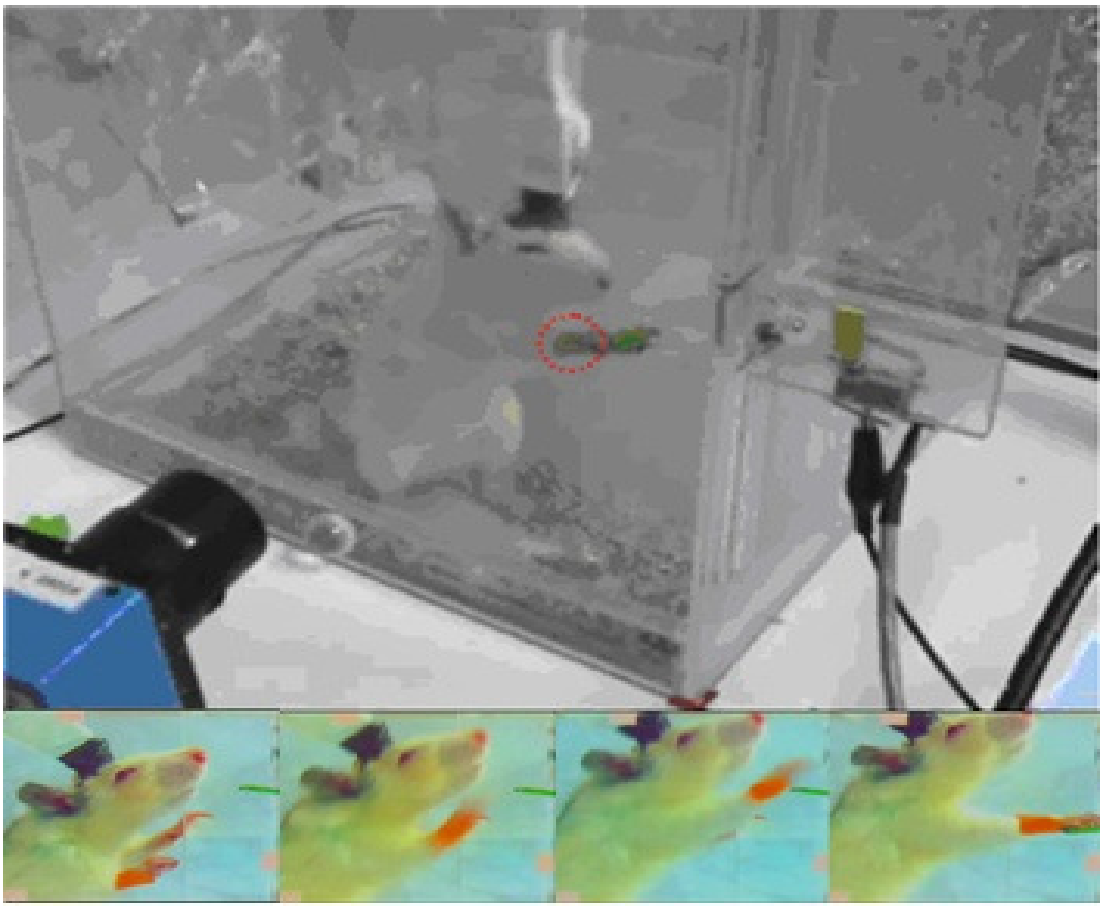


Figure 3.1:ECoG recording during rat movement.

4 Results and Simulation

4.1 Signal Filtering:

The first step and the purpose of spike sorting is the signal preprocessing to prepare the signal in order to simplify the next steps of the processing. At the beginning, a channel is represented with a collection of many states of a group of neighbor neurons with different positions of contacts at the level of the in-out of the Soma, dendrite, axon …etc. According to the points of contacts; the used sensor records a temporal superposition of many neuronal influences on only one channel and it has been. This will be big problem for the application of the brain computer interface. So a signal processing is necessary to get separated information of each neuron.

Firstly, the signal will be filtered with a band pass filter between 300 and 3000Hz to eliminate noises and obtain the original signal containing spikes.

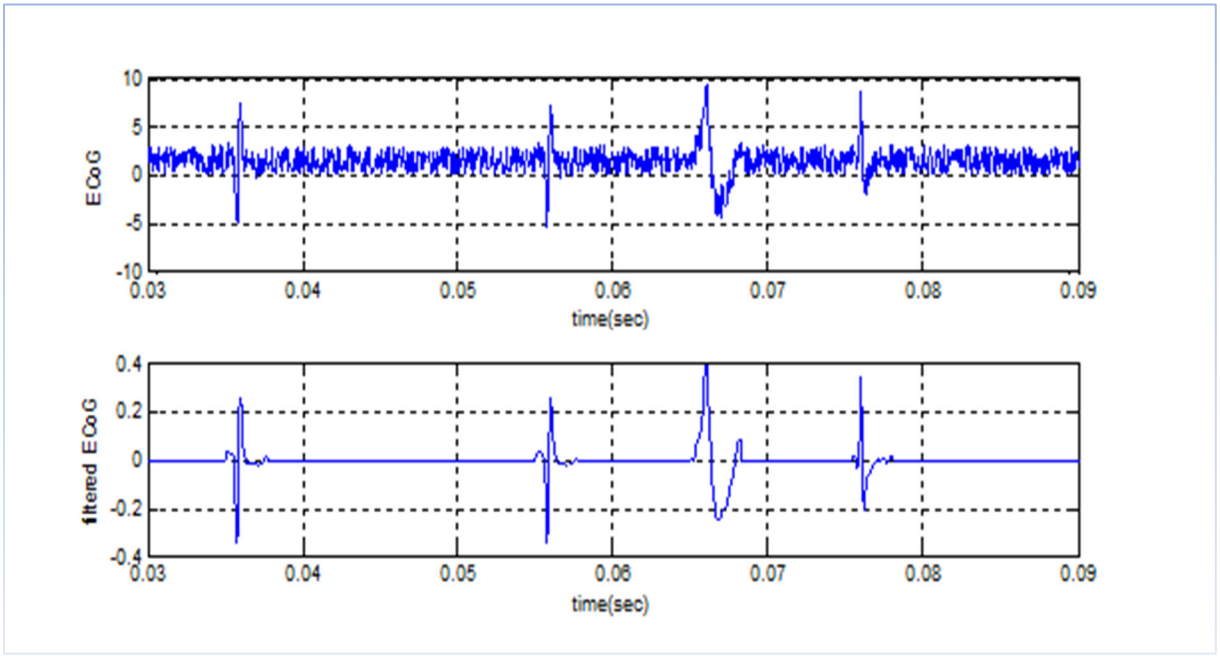


Figure 3.2: ECoG original and filtered.

Figure 3.2 illustrates the original signal with its filtered version with a band pass filter from 300Hz to 3 KHz. It can be seen the difference between the two signals where there is no noise in the filtered signal with a clear spikes shapes.

The filtered signal will be simple for other next steps of the spikes sorting process.

4.2 Spike Detection and Alignment:

At this stage, the recorded signal has already been filtered. For the detection of spikes, we use the NEO operator function (Nonlinear Energy Operator) applied on the signal to make spikes more clear and easy to be localized. Spike shapes became more important energy and with the use of a simple threshold will be detected. The threshold should be chosen depending on the NEO results using specific equations as mentioned in the second chapter. A normalized time window will be centered on the maximal spikes position energy to limit spikes.

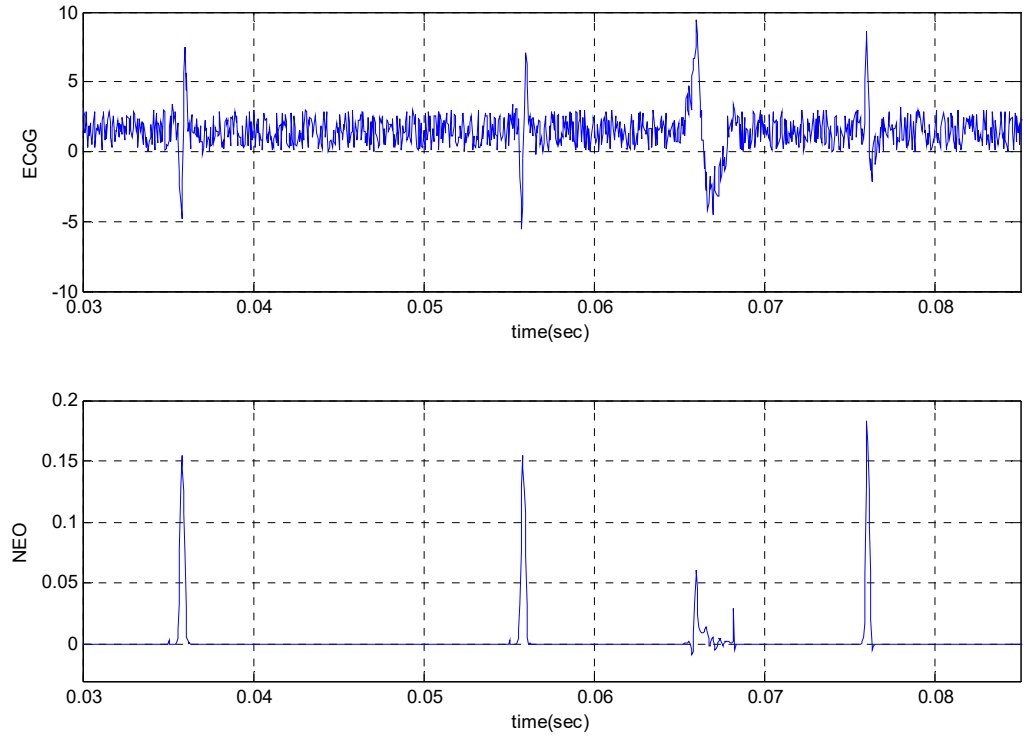


Figure 3.3: Detection using the operator NEO.

NEO coefficient:

The value of the NEO operator $\boldsymbol{\psi}$(x[n]) is calculated to the characteristic spike minimum value using equation 3.1, where $\boldsymbol{\psi}(\boldsymbol{x}[\boldsymbol{n}])$is the value of the NEO operator at the $\boldsymbol{n^{th}}$ time sample, $\boldsymbol{x}[\boldsymbol{n}]$ is the $\boldsymbol{n^{th}}$ sample of the signal:

$$\boldsymbol{\psi}(\boldsymbol{x}[\boldsymbol{n}]) = \boldsymbol{x}^2[\boldsymbol{n}] - \boldsymbol{x}[\boldsymbol{n}-\boldsymbol{1}] \times \boldsymbol{x}[\boldsymbol{n}+\boldsymbol{1}] \qquad (3.1)$$

The resulting parameter $\boldsymbol{\psi}$ (x[n]) gives an estimate of the energy content of the signal.

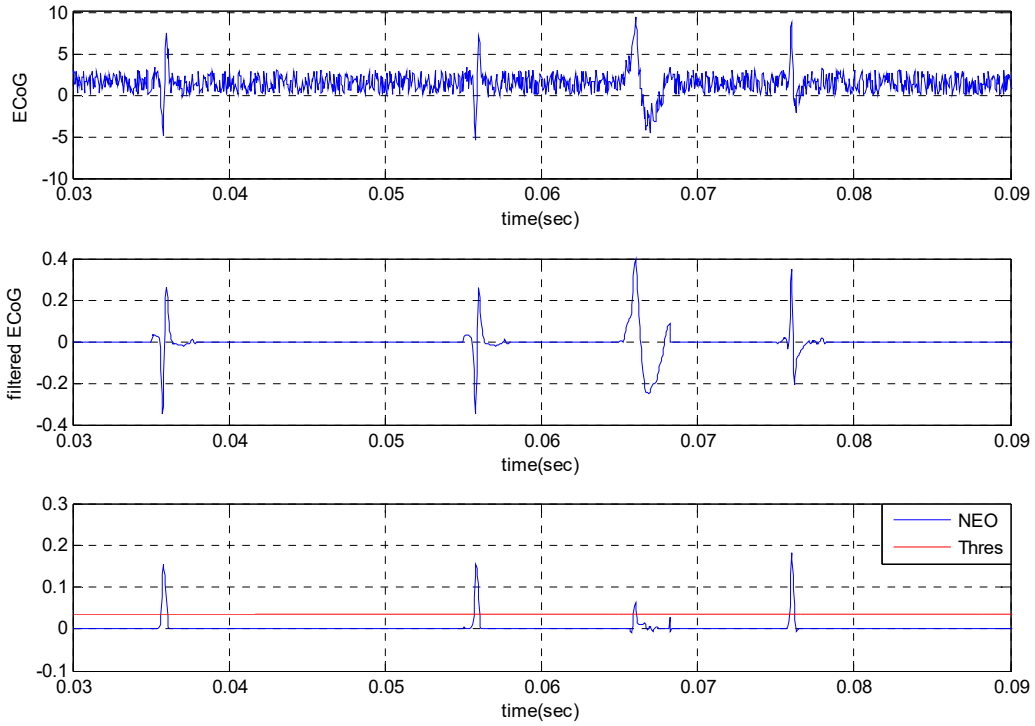


Figure 3.4: Superposition of the NEO with the Threshold.

Figure 3.4 shows the results of the NEO applied on the filtered signal. The energy of the NEO is more important at each spike position. It can be detected easily by the use of a specific threshold calculated using the formula:

$$Thr = C\frac{1}{N}\sum_{n=1}^{N}\boldsymbol{\psi}(\boldsymbol{x}(\boldsymbol{n})) \tag{3.2}$$

Where C is a scaling factor (C=3) and $\boldsymbol{\psi}(\boldsymbol{x}(\boldsymbol{n}))$is the NEO at n^{th} sample time. Figure 3.5 illustrates the superposition of the NEO with the Threshold. Each spike energy crosses the threshold and be identified and localized at the maximal value of the energy. Figure 3.6 demonstrates the localization of three successively crossing of the NEO (in blue) with the threshold (in green) which mean the presence of four spikes temporal successively. Spike centers will be at the maximal value of the NEO between up and down crossing.

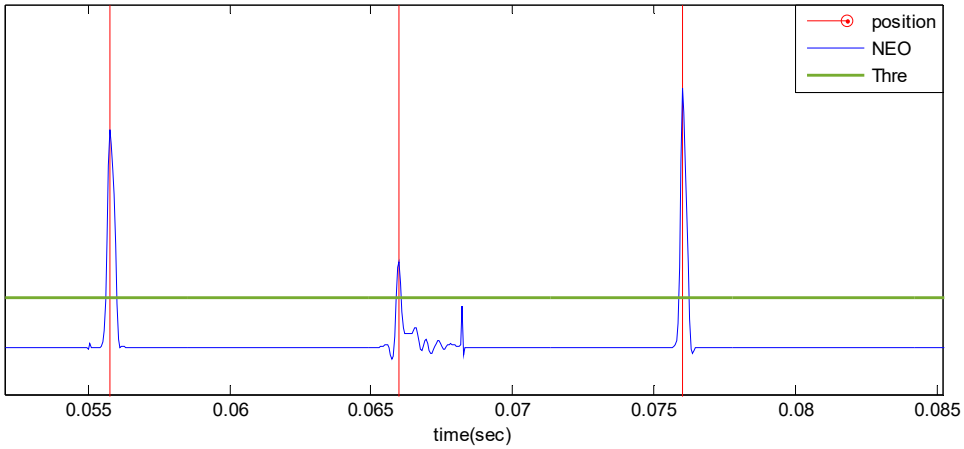


Figure 3.5: Spike time localization with NEO.

Figure 3.7 illustrates the detection of spikes and their position after the application of the threshold on the NEO. It can be seen the good detection of spikes position extracted from the comparison of the NEO to the threshold, where each detected position is superposed on a real spike in ECoG. The spikes positions depend on the maximal value of the NEO instead of the extremes (minima and maxima) of the ECoG, it can be seen also that positions takes different situation on the evolution of the potential. These positions will be the center of the time windows that contain the spikes shapes.

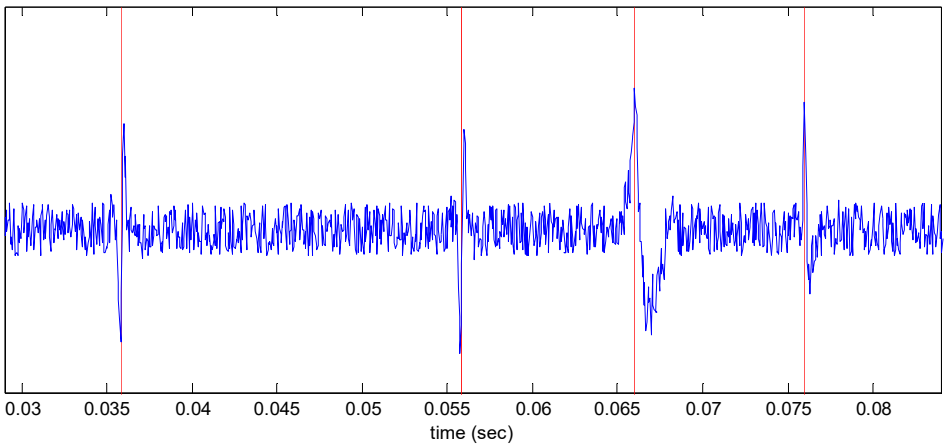


Figure 3.6: Real spikes positions.

4.3 Isolation

This part consists of the isolation of detected spikes where the temporal spikes delimitation of shapes is done. We determine the low and up time windows limit of the spikes shapes. Figure 3.8 summarizes a superposition of detected spike shapes collected together, as it is obvious a three groups of resemble shapes. This opens another work to apply any technic of classification and clustering.

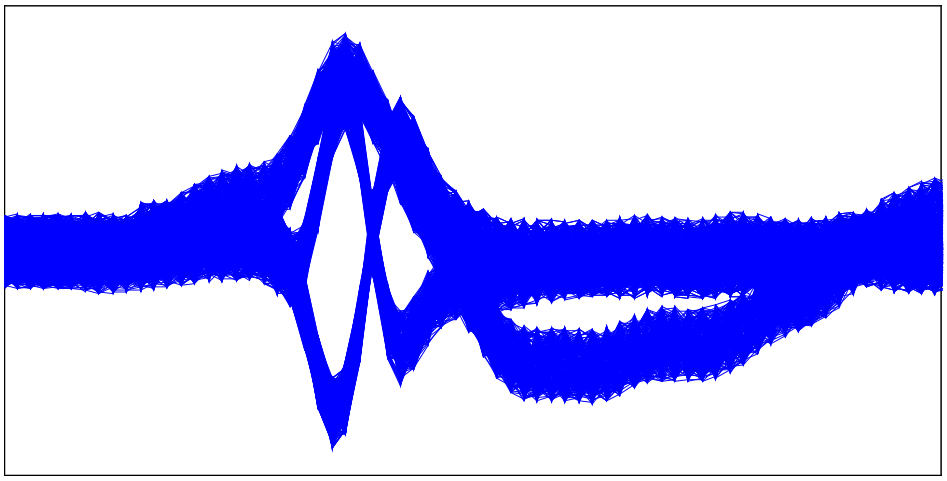


Figure 3.7: Detected spikes shapes.

4.4 Classification

Spike sorting aims to group similar spikes together, based on the assumption that similar spikes originate from the same neuron, so that each cluster represent a neuron (or similar multi-unit). The spikes clustering are based on two principal lows:

-if the spike looks like a spike shape, then it will be ranked with this group of spikes.

-if the spike does not look like any of the spikes shapes, then a new line is recorded in the class matrix.

In figure 3.9, we can see clearly three different colored shapes, each color represents a group of spikes.

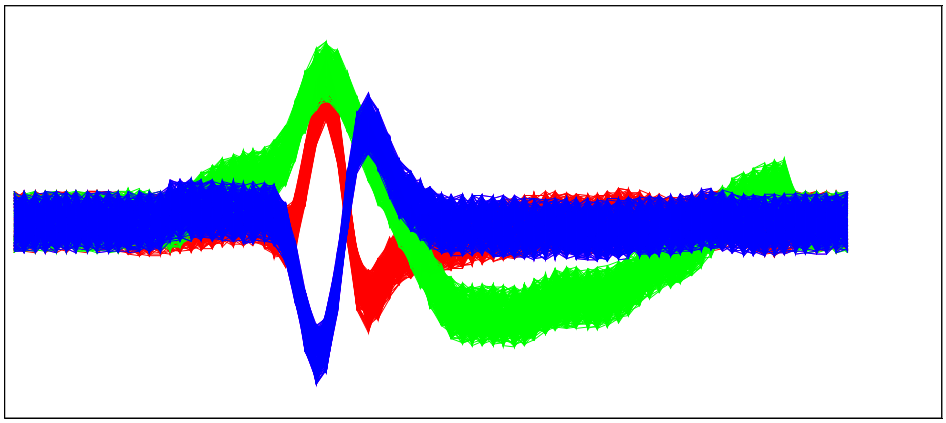


Figure 3.8: Classification of spikes.

4.5 Spikes Representation

The spikes sorting gives 48 binary time series channels of neural firing rate represented by '1' separated by a long silence represented by '0'. The real information corresponding to the hand position (lever) is included in the time between spikes. So a special representation is important for that and also to synchronize the time samples of the hand position with the neural firing rates.

In figure 3.10 is shown different techniques and comparison of continuous spikes train representation. Figure (3.10.A) shows three seconds of the response of a neuron in the inferior temporal cortex recorded while the experiment. (Mention that in the region of cortex where this recording was made are selective for actions and behaviors). A simple way of extracting an estimate of the firing rate from a spike train like this is to divide time into discrete bins of duration t, count the number of spikes within each bin, and divide by t. Figure (3.10.B) shows the approximate firing rate computed using this procedure with a bin size of $100ms$. Note that, with this procedure, the quantity being computed is really the spike-count firing rate over the duration of the bin, and that the firing rate $r(t)$ within a given bin is approximated by this spike-count rate. The binning and counting procedure illustrated in figure (3.10.B) generates an estimate of the firing rate that is a piecewise constant function of time, resembling a histogram. Because spike counts can only take integer values, the rates computed by this method will always be integer multiples of $1/\delta t$, and thus they take discrete values. Decreasing the value of t increases temporal resolution by providing an estimate of the firing rate at more finely spaced intervals of time, but at the expense of decreasing the

resolution for distinguishing different rates. One way to avoid quantized firing rates is to vary the bin size so that a fixed number of spikes appear in each bin. The firing rate is then approximated as that fixed number of spikes divided by the variable bin width. In figure (3.10.C) is shown an approximate of the firing rate determined by sliding a rectangular window function along the spike train with $\delta t = 100ms$and in figure (3.10.D) is the same as in C but with a Gaussian function.

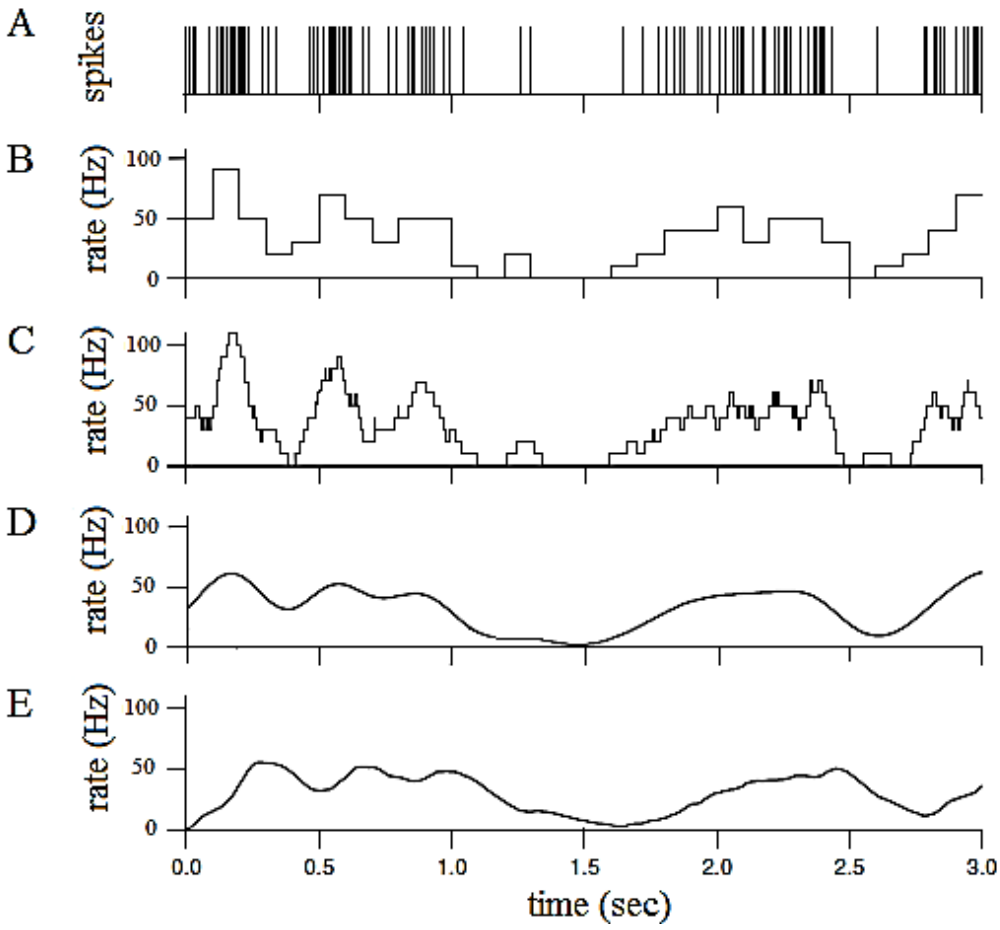


Figure 3.9: Continuous spikes train representation.

5 Conclusion

In this chapter, we have seen an overview on MATLAB software; its definition, uses and MATLAB Simulink. Then, we talked about the experiment done on male Wistar rats, where it has been trained to press a lever to initiate a trial in return for a water reward. At that time; the dataset was collected from the rat's motor cortex obtaining the ECoG. Depending on the ECoG data recorded, we presented the spike sorting results using the MATLAB software applying specific algorithms, such as: filtering, detection and alignment (using NEO coefficient and thresholding), isolation, classification and spikes representation. Finally, all these mentioned steps have been done online.

GENERAL CONCLUSION

General Conclusion

Spike sorting is a very challenging mathematical problem; it is a crucial step to extract information from extra\intracellular recordings. With new recording opportunities provided by the development of new electrodes that allow monitoring hundreds of neurons simultaneously, this problem has attracted the attention of scientists from different fields. It is indeed an interesting problem for researchers working on signal processing, especially those dealing with pattern recognition and machine learning techniques. It is also crucial for neurophysiologists, since an optimal spike sorting can dramatically increase the number of identified neurons and may allow the study of very sparsely firing neurons, which are hard to find with basic sorting approaches.

As the algorithms for spike sorting can be quite complicated and given this can be a difficult and time consuming process, it is worth asking whether it is really necessary to do spike sorting rather than taking all the spikes together, as the lump activity of an unknown number of neurons. The problem is that close-by neurons -picked up by the same electrode - can fire in response to different things. This is the case, for example, in the human or rat hippocampus, where nearby neurons fire to unrelated people in the first case and to distant place fields in the latter. But even when nearby neurons have similar responses, it is important to distinguish them and observe their individual tuning properties, firing characteristics, relationship with other neurons and local field potentials, and so on. One strategy to avoid having to use complex spike-sorting algorithms is to use acute electrodes lowered into the animals' brain during each experiment. Then, the electrode can be placed sufficiently close to a given neuron, decreasing interference of the spikes from others. There are, however, several caveats with this approach:

- First, it introduces a bias towards recording from high firing (and typically less selective) neurons.
- Second, it is possible to observe only one or very few neurons at a time.

The possibility of recording from hundreds or thousands of neurons simultaneously is the dream of any neurophysiologist and a goal that is within reach, as it is now possible to record from hundreds of channels simultaneously. There is clearly a need to develop fully automatic, fast spike-sorting algorithms to deal with such large number of channels and the massive volumes of recorded data. The advantage of using tetrodes is also clear, but current spike

sorting algorithms still use relatively naïve methods to combine the information from different sites. Further developments of spike-sorting algorithms should go together with the optimization of electrode designs with the general goal of maximizing the number of simultaneously recorded and identified neurons.

BIBLIOGRAPHY

Bibliography

[1]: Anupama, H. S., Cauvery, N. K., &Lingaraju, G. M. (2012). Brain computer interface and its types-a study. International Journal of Advances in Engineering & Technology, 3(2), 739.

[2]: DominikZajíček , introduction to brain machine interface, Thesis Assignment, Comenius University in Bratislava,25.10.2011.

[3]: A seminar report ,BRAIN-MACHINE INTERFACE ,SHRIDHAR DUNDAPPA MASAGONDA,VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM, 2009/2010

[4]: Oliver Tonet et al,(2008) critical review and future perspective of non invasive brain machine interfaces, esa research ,272 ,7(2)

[5]: Abdulkader, S. N., Atia, A., &Mostafa, M. S. M. (2015). Brain computer interfacing: Applications and challenges. Egyptian Informatics Journal, 16(2), 213-230.

[6]: Ramadan, R. A., Refat, S., Elshahed, M. A., & Ali, R. A. (2015). Basics of brain computer interface. In Brain-Computer Interfaces(pp. 31-50). Springer, Cham.

[7]: S.bilal et al, separation des spikes (2017),thesis, university of Ghardaia

[8]: GRAIMANN, Bernhard, ALLISON, Brendan, et PFURTSCHELLER, Gert. Brain–computer interfaces: A gentle introduction. In : Brain-Computer Interfaces. Springer, Berlin, Heidelberg, 2009. p. 1-27.

[9]: Marie Hélène Bekaert*et al ,les interfaces cerveau machine pour la palliation du handicap moteur sévère 20/11/2009.

[10]: Ramadan, R. A., &Vasilakos, A. V. (2017). Brain computer interface: control signals review. Neurocomputing, 223, 26-44.

[11]:Gibson, Sarah Paige,Neural Spike Sorting in Hardware: From Theory to Practice,thesis,01-01-2012,University of California Los Angeles

[12]: Bestel, R., Daus, A. W., &Thielemann, C. (2012). A novel automated spike sorting algorithm with adaptable feature extraction. Journal of neuroscience methods, 211(1), 168-178.

[13]: QUIROGA, Rodrigo Quian. Spike sorting. Current Biology, 2012, vol. 22, no 2, p. R45-R46.

[14]: NATAL, Leonardo Braga. Dedicated Processor for Real-time Spike Sorting. 2016.

[15]: Charles K. Alexander and Matthew N.O Sadiku, "Electric Circuits", 2nd edition, McGraw-Hill 2004.

[16]: Hayes, (Monson H.), Schaum's outline of theory and problems of digital signal processing / Monson H. Hayes. p. cm. — (Schaum's outline series), 1999.

[17]: www.perso.univ-lemans.fr, "Phy308aA_5_C1.pdf", 21/05/2018

[18]: HAMILA, Ridha, ASTOLA, Jaakko, CHEIKH, F. Alaya, etal.Teager energy and the ambiguity function. IEEE Transactions on Signal Processing, 1999, vol. 47, no 1, p. 260-262.

[20]:Kyung Hwan Kim and Sung June Kim, Neural Spike Sorting Under Nearly 0-dB Signal-to-NoiseRatio Using Nonlinear Energy Operator and ArtificialNeural-Network Classifier, IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, VOL. 47, NO. 10, OCTOBER 2000

[21]: RIZK, Michael et WOLF, Patrick D. Optimizing the automatic selection of spike detection thresholds using a multiple of the noise level. Medical&biological engineering &computing, 2009, vol. 47, no 9, p. 955-966.

[22]: Med BiolEngComput, Optimizing the Automatic Selection of Spike Detection Thresholds Using a Multiple of the Noise Level, Published online 2009 Feb 10.

[23]: S. Gibson, J. W. Judy and D. Markovic."Comparison of spike-sorting algorithms for future hardware implementation," IEEE Engineering inMedicine and Biology Society, pp. 5015 - 5020, Aug. 2008.

[24]: MALIK, Muhammad H., SAEED, Maryam, et KAMBOH, Awais M. Automatic threshold optimization in nonlinear energy operator based spike detection. In : Engineering in

Medicine and Biology Society (EMBC), 2016 IEEE 38th Annual International Conference of the. IEEE, 2016. p. 774-777.

[25]: DO, Anh Tuan et YEO, Kiat S. A hybrid NEO-based spike detection algorithm for implantable brain-IC interface applications. In : Circuits and Systems (ISCAS), 2014 IEEE International Symposium on. IEEE, 2014. p. 2393-2396.

[26]: BERGHEIM, Thomas Stianet SKOGVOLD, ArveAleksanderNymo. Parallel Algorithms for Neuronal Spike Sorting. 2011. Thèse de maîtrise. Institutt for datateknikkoginformasjonsvitenskap.

[27]: Wild, J., Prekopcsak, Z., Sieger, T., Novak, D., &Jech, R. (2012). Performance

comparison of extracellular spike sorting algorithms for single-channel recordings.Journal of neuroscience methods, 203(2), 369-376.

[28]: MIN, Byoung-Kyong, CHAVARRIAGA, Ricardo, et MILLÁN, José del R. Harnessing Prefrontal Cognitive Signals for Brain–Machine Interfaces. Trends in biotechnology, 2017, vol. 35, no 7, p. 585-597.