

# POLITECNICO DI TORINO

Master's Degree in Biomedical Engineering



Master's Degree Thesis

## Towards selective recording in the Peripheral Nervous System using cuff electrodes

Supervisors

Prof. DANILO DEMARCHI

Prof. TIMOTHY CONSTANDINOU

Prof. MAURIZIO MAGARINI

Dr. ADRIEN RAPEAUX

Candidate

FABIANA DEL BONO

October 2020



# Summary

Neural recording in the peripheral nervous system implies to find a balance between selectivity of the recording and invasiveness of the electrodes used. Cuff electrodes consist of cylindrical structures made of an insulating biocompatible material with metal contacts on the inner face, to be placed around the nerve surface. Among the available categories, they are the least invasive and have shown the best long-term implant stability, as they allow to record electroneurograms without compromising the integrity of the nerve structure. However, the identification of the active fibres inside the nerve is extremely complicated if compared with more invasive alternatives, and requires powerful classification algorithms to interpret the signals, achieving selectivity. Although different designs have been investigated in research, setting the optimal configuration of the contacts is not trivial, and implies a complexity trade-off between the cuff and the processing units. The purpose of this work is to provide a tool for simulating the collection of electroneurograms from the peripheral nerve, thus enabling to test different configurations of cuff electrodes and signal processing algorithms to increase selectivity. The project required to find the best simulation strategy and led to implementing a Multiphysics model from scratch, capable of producing simulated electroneurograms and making them available for processing. The resulting tool allows an integration of the single neurons' simulations, obtained with specific software, within a Multiphysics model of the nerve coupled with the cuff electrode. The proposed setup allows testing of user-defined electrode designs, exploiting a parametrization of their features. The signals sensed by each contact can be exported and processed, enabling the implementation of algorithms for activity recognition. Performances on the recording of evoked potentials showed a match with experimental results, and the developed strategy leaves room for increasing complexity according to the user's needs. This will lead, in the long term, to streamline the design of stable and selective peripheral nerve interfaces and to optimise animal tests limiting the use of resources.



# Table of Contents

<b>List of Tables</b>	VII
<b>List of Figures</b>	IX
<b>Acronyms</b>	XIII
<b>1 Motivation</b>	1
<b>2 Background</b>	3
2.1 Introduction to Peripheral Nervous System . . . . .	3
2.1.1 Cells in the Nervous System . . . . .	4
2.1.2 The neuron . . . . .	5
2.1.3 Peripheral Nerves . . . . .	9
2.2 Introduction to Peripheral Nerve Interfaces . . . . .	10
2.2.1 The electroneurogram . . . . .	10
2.2.2 Electrodes for Peripheral Nerves . . . . .	11
2.2.3 Cuff electrodes . . . . .	14
<b>3 State of the Art</b>	17
3.1 Selective recording strategies . . . . .	17
3.1.1 Improving selectivity with the processing hardware . . . . .	17
3.1.2 Improving selectivity with processing algorithms . . . . .	18
3.1.3 The need for simulations . . . . .	20
3.2 Peripheral Nerve simulations . . . . .	21
3.2.1 Nerve characterisations . . . . .	21
3.2.2 Overview of the simulation approaches . . . . .	24
3.2.3 Axon models . . . . .	24
3.2.4 Extracellular medium simulation . . . . .	26
3.2.5 Simulations with multiple axons . . . . .	27
3.2.6 FEM-based approaches . . . . .	27
3.2.7 Whole-nerve models . . . . .	28

<b>4</b>	<b>Tools and Method</b>	<b>30</b>
4.1	Overview of the model . . . . .	30
4.2	Simulation of the axons . . . . .	31
4.2.1	The NEURON Simulation Environment . . . . .	31
4.2.2	Modeling the axons . . . . .	32
4.3	Multiphysics Model for PN Recording . . . . .	36
4.3.1	Modeling the system with COMSOL Multiphysics . . . . .	37
4.3.2	Cuff electrode geometric model . . . . .	38
4.3.3	Saline medium and nerve segment model . . . . .	42
4.3.4	Setting the materials . . . . .	46
4.3.5	Physics . . . . .	48
4.3.6	Study configurations and generation of the results . . . . .	51
<b>5</b>	<b>Results</b>	<b>56</b>
5.1	Output of the Neuron simulations . . . . .	56
5.2	Preliminary simulations . . . . .	58
5.2.1	Effect of fibre diameter . . . . .	58
5.2.2	Effect of the thickness of the cuff . . . . .	60
5.2.3	Effect of the mesh . . . . .	62
5.2.4	Effect of the nervous tissue model . . . . .	63
5.2.5	Effect of ground dimension . . . . .	64
5.3	Single axon models . . . . .	67
5.3.1	Effect of the cuff geometry . . . . .	67
5.3.2	Effect of the fibre position . . . . .	71
5.4	Multiple axons models . . . . .	76
5.4.1	Simulations with 2 fibres . . . . .	77
5.4.2	Simulations of CAPs . . . . .	80
5.5	Reliability of the results . . . . .	82
<b>6</b>	<b>Conclusions</b>	<b>85</b>
<b>A</b>	<b>Scripts</b>	<b>86</b>
A.1	NEURON . . . . .	86
A.1.1	<i>mosinit.hoc</i> . . . . .	86
A.1.2	<i>MRGaxon.hoc</i> . . . . .	87
A.1.3	<i>Vec_generationLoop</i> . . . . .	92
A.1.4	<i>currentConversion.hoc</i> . . . . .	92
A.1.5	<i>fileCreationLoop.hoc</i> . . . . .	93
A.2	COMSOL . . . . .	94
A.2.1	Initialisation of counters . . . . .	94
A.2.2	Import currents from Neuron . . . . .	95

A.2.3	Creation of the axons . . . . .	96
A.2.4	Create electrode selection . . . . .	101
A.2.5	Insert point current sources . . . . .	102
A.2.6	Export voltage on the contacts . . . . .	103
A.3	MATLAB . . . . .	106
A.3.1	NOR current plots . . . . .	106
A.3.2	Voltage plots for MEC . . . . .	107
A.3.3	Voltage plots for MCC . . . . .	109
A.3.4	Computation of FWHM . . . . .	114
<b>Bibliography</b>		<b>116</b>

# List of Tables

2.1	Fibres in the PNS and related conduction velocities, adapted from .	9
3.1	Summary of the most popular processing algorithms for selective recording . . . . .	21
3.2	Fiber counts in the rat sciatic nerve, adapted from [31] . . . . .	22
3.3	Adapted from [34]. . . . .	23
3.4	Electrical properties of fluids and extracellular volumes in the nerve. Adapted from [34]. . . . .	24
4.1	Original settings for MRG axon . . . . .	34
4.2	Redefinition of the original topological parameters as a function of the computed node number. . . . .	34
4.3	List of the axon geometrical parameters computed for a 6 cm nerve fibre . . . . .	35
4.4	Table of the geometrical parameters used in COMSOL for the definition of the cuff electrode . . . . .	40
4.5	List of the dependant parameters and the expressions for calculating their value starting from user-defined ones . . . . .	41
4.6	Functions for the calculation of the contacts positions along the z axis and the circumference, respectively. . . . .	41
4.7	Parameters used for the definition of the nerve structure and the saline medium . . . . .	43
4.8	List of the NEURON parameters imported in COMSOL . . . . .	44
4.9	PDMS electrical properties . . . . .	47
4.10	Platinum electrical properties . . . . .	47
4.11	0.9% NaCl based saline solution electrical properties . . . . .	48
4.12	Nerve tissue electrical properties at 1kHz . . . . .	48
5.1	Computed CVs and current amplitudes for the 9 types of fibres simulated within NEURON. . . . .	58
5.2	Starting parameters for the preliminary simulations . . . . .	59



5.3	Peak-to-peak amplitudes on the electrode rings and their average in the recording from nerves with different diameters. . . . .	60
5.4	Peak-to-peak amplitude on a 1 mm fibre with 1.5 mm diameter cuffs with different thickness. . . . .	61
5.5	Peak-to-peak amplitude on a 2 mm fibre with 3 mm diameter cuffs with different thickness. . . . .	62
5.6	Peak-to-peak amplitude on a 1 mm fibre with different mesh settings	63
5.7	Peak-to-peak amplitude on a 2 mm fibre with different mesh settings	64
5.8	Peak-to-peak amplitude on a 1 mm nerve with different different definitions of the nerve tissue electrical properties . . . . .	65
5.9	Peak-to-peak amplitude on a 1 mm nerve with different different definitions of the nerve tissue electrical properties . . . . .	66
5.10	Peak-to-peak amplitude on a 1 mm nerve with different diameters of the cylinder set as ground. . . . .	67
5.11	Signals recorded by varying the the dimensions of the outer edges of the cuff (distance between the outer electrodes and the ends of the tube) keeping IED constant. . . . .	68
5.12	Effect of the edge dimension on the waveform parameters of the average signals sensed on the different rings . . . . .	71
5.13	Effect of the position of the nerve fibre across the nerve section on the peak-to-peak potential amplitudes and on the waveform FWHM, in the case of axons located centrally and towards the surface. Signals recorded with MECs. . . . .	72
5.14	Peak-to-peak values, average and FWHM for the potential waveform from a 5.7 $\mu\text{m}$ centrally-located fibre, obtained different electrode types. . . . .	75
5.15	Peak-to-peak values, average and FWHM for the potential waveform from a 5.7 $\mu\text{m}$ surface-located fibre, obtained different electrode types.	76
5.16	Diameters and coordinates of the fibres created for the simulation of a CAP. . . . .	82

# List of Figures

2.1	Overview of the nervous system . . . . .	4
2.2	Anatomy of a neuron . . . . .	5
2.3	Plot of an action potential . . . . .	7
2.4	Comparison between myelinated and unmyelinated fibres . . . . .	8
2.5	Structure of a peripheral nerve . . . . .	9
2.6	Shape of a compound action potential (CAP) . . . . .	11
2.7	Selectivity and invasiveness of the existing peripheral nerve electrode categories, compared with the ideal case. Image from [2]. . . . .	12
2.8	Example of a cuff electrode . . . . .	14
2.9	C-shaped and Spiral cuff electrode designs . . . . .	15
2.10	Multielectrode cuff (left) and multicontact cuff (right) designs. Credits: M. Schuettler . . . . .	16
3.1	Tripolar configuration and its variants [2] . . . . .	18
3.2	Anatomy of a rat sciatic nerve [30]. . . . .	22
3.3	Currents in a compartmental model of a neuron (3.3a) and electric representation of a neuron (3.3b) according to the model. From [38]. . . . .	25
4.1	Graphic representation of compartments in the MRG axon model [33]. . . . .	33
4.2	Currents for all the nodes of Ranvier in a 16 micrometers axon . . . . .	37
4.3	Reference frame for the cuff and the nerve models . . . . .	39
4.4	Example of a MCC created in COMSOL. . . . .	43
4.5	Nerve, saline and axon geometries, in the case of a 5.7 and 16 $\mu\text{m}$ diameter axons. It is possible to notice the difference in spacing between the NORs. . . . .	45
4.6	Nerve and cuff geometries, coupled together. The cylinder representing the saline solution is hidden. . . . .	46
4.7	Nerve electrical properties from 0 to 100kHz, retrieved from [53] . . . . .	49
4.8	Conditions imposed on the physics of the model, highlighted in yellow. In 4.8a, the bases set as electric insulation are shown, while 4.8b represents the surface of the saline medium declared as ground. . . . .	51

4.9	Example of slice plots, representing the propagation of an action potential in a 1-fibre nerve model. . . . .	53
4.10	Example of volume plots on the rings, showing the propagation of an action potential as read by the electrode. . . . .	54
4.11	Example of volume and slice plots of the nerve and electrode rings, showing the propagation of an action potential. . . . .	55
5.1	Examples of the plot of the currents in the central Node of Ranvier, the the previous tenth NOR and the following tenth NOR, simulated in the NEURON software and used for computing the CV and the current amplitude. . . . .	57
5.2	Potentials recorded from nerves with different diameters. . . . .	59
5.3	Potential read on the 4 rings of a MEC on a 1 mm fibre with 1.5 mm diameter cuffs with different thickness. . . . .	60
5.4	Potential read on the 4 rings of a MEC on a 2 mm fibre with 3 mm diameter cuffs with different thickness . . . . .	61
5.5	Potential on a 1 mm fibre with different mesh settings . . . . .	62
5.6	Potentials on a 2 mm fibre with different mesh settings . . . . .	63
5.7	Signals on a 1 mm nerve with different definitions of the nerve tissue electrical properties . . . . .	64
5.8	Peak-to-peak amplitude on a 1 mm nerve with different different definitions of the nerve tissue electrical properties . . . . .	65
5.9	Signals on a 2 mm nerve with different diameters of the cylinder set as ground. . . . .	66
5.10	Signals recorded by varying the the dimensions of the outer edges of the cuff (distance between the outer electrodes and the ends of the tube) keeping IED constant . . . . .	69
5.11	Effect of the edge dimension on the shape and the relative amplitude of the average signals sensed on the rings . . . . .	70
5.12	Section of the nerve showing different positions of the axon . . . . .	72
5.13	Effect of the position of the nerve fibre across the nerve section on the potential amplitudes recorded with MECs in the case of axons located centrally and towards the surface. . . . .	73
5.14	Potentials from a 5.7 $\mu\text{m}$ centrally-located fibre averaged from the 4 rings, obtained different electrode types. . . . .	74
5.15	Potential values on each ring from a 5.7 $\mu\text{m}$ centrally-located fibre, obtained different electrode types. . . . .	74
5.16	Potentials from a 5.7 $\mu\text{m}$ surface-located fibre averaged from the 4 rings, obtained different electrode types. . . . .	75
5.17	Potential values on each ring from a 5.7 $\mu\text{m}$ surface-located fibre, obtained different electrode types. . . . .	76

5.18	Plots on the potentials on the 1st and 5th contact of a 4-rings MCC. Nerve with central 5.7 $\mu\text{m}$ fibre and surface 16 $\mu\text{m}$ fibre. . . . .	77
5.19	Plots on the contact-averaged potentials on the first and last ring of the MCC. Nerve with central 5.7 $\mu\text{m}$ fibre and surface 16 $\mu\text{m}$ fibre. . . . .	78
5.20	Plots on the potentials on the 1st and 5th contact of a 4-rings MCC. Nerve with central 5.7 $\mu\text{m}$ fibre and surface 10 $\mu\text{m}$ fibre. . . . .	79
5.21	Plots on the contact-averaged potentials on the first and last ring of the MCC. Nerve with central 5.7 $\mu\text{m}$ fibre and surface 10 $\mu\text{m}$ fibre. . . . .	79
5.22	Plots on the potentials on the 1st and 5th contact of a 4-rings MCC. Nerve with central 16 $\mu\text{m}$ fibre and surface 5.7 $\mu\text{m}$ fibre. . . . .	80
5.23	Plots on the contact-averaged potentials on the first and last ring of the MCC. Nerve with central 16 $\mu\text{m}$ fibre and surface 5.7 $\mu\text{m}$ fibre. . . . .	81
5.24	Plots on the potentials on the 1st and 5th contact of a 4-rings MCC. Nerve with central 10 $\mu\text{m}$ fibre and surface 5.7 $\mu\text{m}$ fibre. . . . .	81
5.25	Plots on the contact-averaged potentials on the first and last ring of the MCC. Nerve with central 10 $\mu\text{m}$ fibre and surface 5.7 $\mu\text{m}$ fibre. . . . .	82
5.26	Section of the nerve model for CAP simulations, showing the fibre distribution. . . . .	83
5.27	Example of a CAP simulated within the simulation environment created. . . . .	84



# Acronyms

**ANN**

Artificial Neural Network

**AP**

Action potential

**BNB**

Blood nerve barrier

**CAP**

Compound action potential

**EMG**

Electromyogram

**FEM**

Finite Element Method

**IED**

Inter-electrode distance

**MCC**

Multicontact cuff

**MEC**

Multielectrode cuff

**NOR**

Node of Ranvier

**PCS**

Point current source

**PNI**

Peripheral nerve interface

**PNS**

Peripheral Nervous System

**SNR**

Signal-to-noise ratio

# Chapter 1

## Motivation

One of the main research trends of recent decades in the biomedical field is focused on the creation of technology that interacts with the nervous system to solve various pathological dysfunctions. These systems, generally having the purpose of restoring a loss of functionality that occurs in the case of congenital pathologies, stroke or trauma improving the patient's quality of life, need to interact with the physiology of the system, exploiting its characteristics of electrical excitability. Consequently, technologies have been implemented that can record the signals of the nervous system or evoke them.

The world of so-called neuroprostheses that results from the implementation of these technologies is extremely wide. Stimulators are used in bladder control, or vagus nerve and auditory nerve stimulation, in which they are able to remodel nervous activity, as well as in functional electrical stimulation, where they can replace lost motor function.

The recording of signals from the nervous system, for neuroprosthetic applications, is used to control the stimulation enabling the creation of closed-loop systems, which include a feedback mechanism to properly calibrate the stimulation parameters in real-time.

In recent times, new trends are also emerging regarding the collection of signals from the nervous system, such as those for the control of bionic prostheses and brain-computer interfaces. These systems have the common goal of allowing the patient to perform actions through the direct exploitation of signals which, under normal conditions, would be used to perform the same function. Neuroprostheses can be applied to both the central nervous system and the peripheral nervous system. Generally, the interaction with the peripheral nervous system should be preferred, if possible. The first advantage is that the patient is subject to less risky surgeries when compared to those on the brain and spinal cord. Furthermore, nerve signals have better interpretability, since, although their generation starts from similar cellular mechanisms, the coding in the nerve is simpler, the number



of sources is less and the level of complexity, consequently, is lowered. Finally, especially in motor functionality, the nerve conveys signals that do not involve central control, such as sensory-motor reflexes. These mechanisms would not be detectable with neural interfaces located in the central nervous system, while this is possible if the interface is located in the peripheral nervous system, through which both sensory stimulation and motor response are transmitted [1].

In any case, the control of the peripheral nervous system is strategic, as it represents, given its function of transporting information and control signals, an attractive therapeutic target in case of injury or disease [2].

For this purpose, the interfaces with the peripheral nervous system should be stable and allow chronic application. However, the only type of interface that has shown long-term stability is the so-called cuff electrode, which nevertheless has very little selectivity. In particular, the recordings of neural signals using this type of electrode are affected by high levels of noise and require a large amount of processing in order to be interpreted. Nowadays, several research groups are working on acquisition systems and processing algorithms to make it possible to obtain selective recordings with cuff electrodes. In these researches, it is very common to conduct animal experiments. To optimize performance without, however, incurring excessive resource consumption, the ideal would be to have models available on which to test different recording and interpretation setups, to then be replicated under experimental conditions. The goal of the work described in this thesis is precisely to provide a flexible tool that allows the activity of a peripheral nerve to be replicated and recorded using cuff electrodes *in silico*, making the signals available for subsequent processing.

The text is organized in a 'Background' chapter, which describes the notions necessary for understanding the work done, in particular those concerning the functioning of the nervous system and the existing neural interfaces. A chapter on the 'State of the Art' follows, in which the existing methods for selective recording and the results of research already carried out on the creation of computational nerve models are analyzed. In the chapter 'Tools and Method', the implemented system for the simulation of extraneural signals, the main topic addressed during the work, is described. Finally, in the 'Results' chapter, some examples of the application of the model and the main observations deriving from them are presented.

# Chapter 2

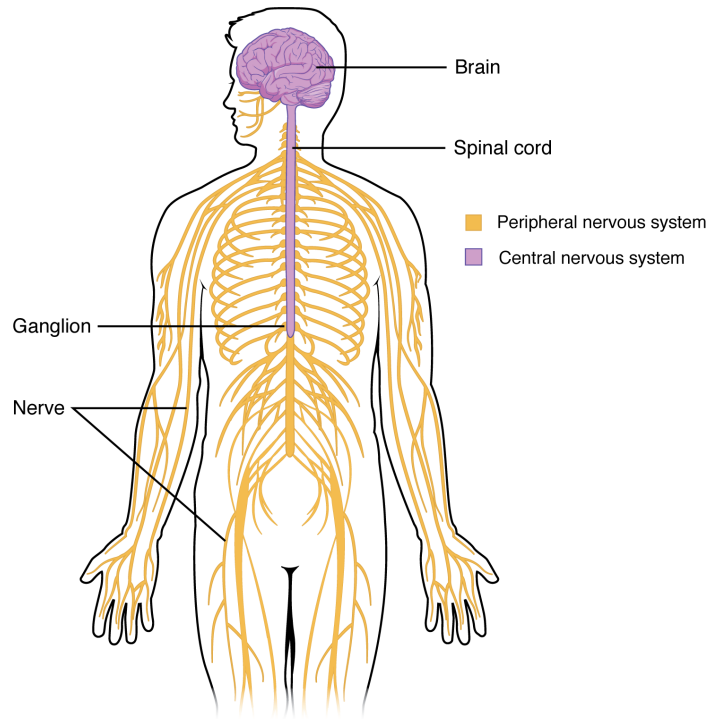
## Background

### 2.1 Introduction to Peripheral Nervous System

The nervous system is the set of organs and structures which is responsible for controlling vital functions and interacting with the external environment, through a constant collection of information, their integration, processing, and the generation and implementation of a response through coordinated impulses that induce activity or functional changes in the parts of the organism. It is divided into two subsystems, the Central Nervous System (CNS) and the Peripheral Nervous System (PNS), shown in figure 2.1. CNS is composed of an axial part (spinal cord and brainstem) and a supraxial part (brain and cerebellum). Its main function is to integrate and process the information received through the sense organs, and to generate responses, as well as to coordinate movement, and to be deputy to memory and language.

PNS consists essentially of nerves, of which 12 pairs are cranial nerves, partly sensory and partly motor, and 31 pairs are sensory-motor spinal nerves. Its function is to rapidly convey signals between the central nervous system and the target organ [3]. A nerve is a set of fibers that carry signals encoding information, called afferent, if this is transported from the central system to the target organs, or efferent, in the opposite case.

The peripheral system can consequently be divided into its afferent and efferent components according to the direction of the information flow. Afferent information can be somatic, sensory or visceral. An afferent is said to be somatic if it comes from skin, muscles or joints, sensory if it comes from the sense organs (sight, hearing, smell, taste and balance), visceral if it comes from the internal environment. Efferent information, on the other hand, modulates the activity of the effector organs, i.e. muscles, internal organs and glands. PNS can also be divided into autonomic and somatic nervous system. The autonomic nervous system has the function of



**Figure 2.1:** Overview of the nervous system

regulating the activity of organs and structures that are not under voluntary control and is in turn divisible into sympathetic and parasympathetic, with opposite effects on the organs. The enteric nervous system, an autonomous nervous network of the gastrointestinal tract, is considered in some cases as part of the autonomic nervous system, in some others as independent. Actually, it functions independently from the rest of the nervous system, but is in communication with the autonomic component. The somatic nervous system, instead, basically includes the voluntary and sensory division, that is, it deals with conveying information to and from the skin and muscles.

### 2.1.1 Cells in the Nervous System

The functional unit of the nervous system is the neuron. Neurons are excitable cells, namely capable of generating action potentials, which are ultimately the electrical signals consisting in the variation of the electrical potential of the neuron's cell membrane, thanks to which information is encoded and transmitted to the internal nervous system. Neurons actually make up a very small part of nerve tissue, about 10% [4].

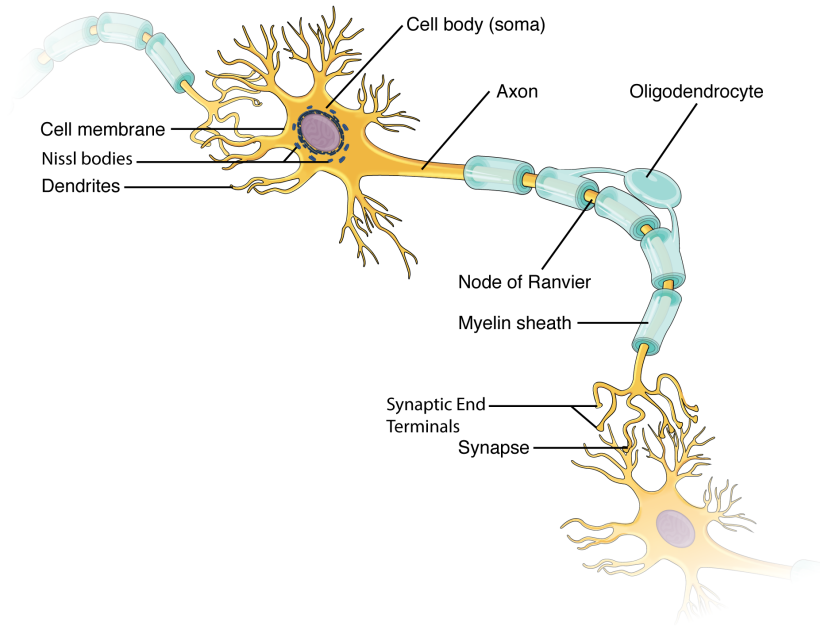
The remaining part, approximately the 90%, is made up of glial cells. Glia means

"glue", and is composed of cells with collateral functions, that are necessary for the structural and metabolic support of neurons, allowing them to perform their functions [4]. Glial cells, specifically, are astrocytes, oligodendrocytes and radial glia in the CNS, and Schwann cells, satellite cells and enteric glial cells in the PNS.

### 2.1.2 The neuron

The anatomy of the neuron is illustrated in the figure 2.2.

The neuron comprises a cell body, called soma, and two types of processes (or



**Figure 2.2:** Anatomy of a neuron

neurites), namely multiple dendrites and a single axon.

The incoming signals reach the neuron from adjacent neurons through so called synapses, which terminate mostly at the level of the dendrites, to a lesser extent at the cell body. The output signal, instead, travels through a single extension that comes out of the neuron, the axon. The signal normally propagates only in an antegrade direction, from the axon of one neuron to the dendrites of the next one.

#### Excitable cells

Neurons are excitable cells. Across their cell membrane, an electrical potential is established. This happens to a certain extent in almost all the cells of the organism,

due to the presence of ions in different concentrations. However, in neurons, this property is emphasized and of fundamental importance, because it is at the basis of their signalling mechanism.

The cell membrane is selectively permeable to ions. Ions diffuse between the extracellular fluid and the intracellular fluid (cytoplasm) as allowed according to the concentration gradient, causing charge imbalances (that generate a "diffusion potential"). This happens until the charge is big enough to prevent further diffusion of the ions. This way, an electrochemical equilibrium is achieved. Each ion has its own equilibrium potential, measured across the cell membrane from the outside to the inside of the cell: for potassium it is -94 mV for sodium 61 mV[5]. The equilibrium potential, namely the one necessary to prevent the net diffusion of an ion across the membrane according to the concentration gradient, is given by the Nernst equation, which takes into account the relationship between the concentrations of the ion and the charge of the same.

Nernst equation allows to calculate the membrane equilibrium potential  $V_m$  for a specific ion  $I$  as:

$$V_{m,I} = \frac{61}{z} \log \frac{[I]_{in}}{[I]_{out}}$$

where  $z$  is the ion charge and  $[I]$  is the ion concentration. However, the cell membrane is permeable to many ions, not just one. In this case, the Goldman-Hodgkin-Katz equation is used, which is a generalization of the Nernst one and derives from it.

$$V_m = \frac{61}{z} \log \frac{P_{I1}[I_1]_{in} + P_{I2}[I_2]_{in} + \dots + P_{In}[I_n]_{in}}{P_{I1}[I_1]_{out} + P_{I2}[I_2]_{out} + \dots + P_{In}[I_n]_{out}}$$

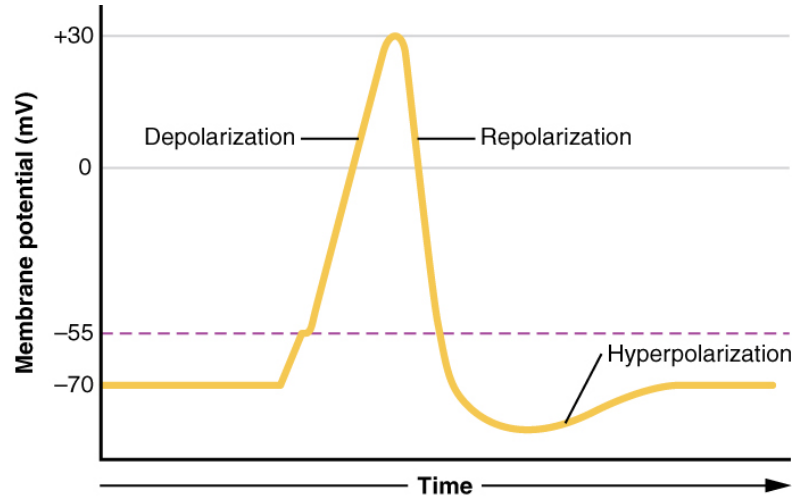
In this case, the weight of each ion on the final voltage is given by the permeability  $P$  that the membrane has, relative to that ion.

The excitability of nerve cells is due to the ability to vary the membrane permeabilities [5]. A contribution to the generation of this potential is the presence of membrane pumps that actively transport ions, helping to establish the negative membrane potential.

### **The action potential**

The electrical impulse in nerve cells is called action potential (AP) and consists of a sudden change in the membrane potential generated by the changes in permeability, and therefore conductance, for each ion, causing a flux of charges. It consists of a depolarization, that is a passage from negative to positive membrane potential, and subsequent repolarization. The fundamental role in this process is played by

the presence of voltage-dependent channels, which modify their permeability to ions according to the voltage condition. Oversimplifying, the sodium channels start the process, the potassium channels make it end quickly, with a slight delay in activation. The beginning of the action potential is generated by a positive



**Figure 2.3:** Plot of an action potential

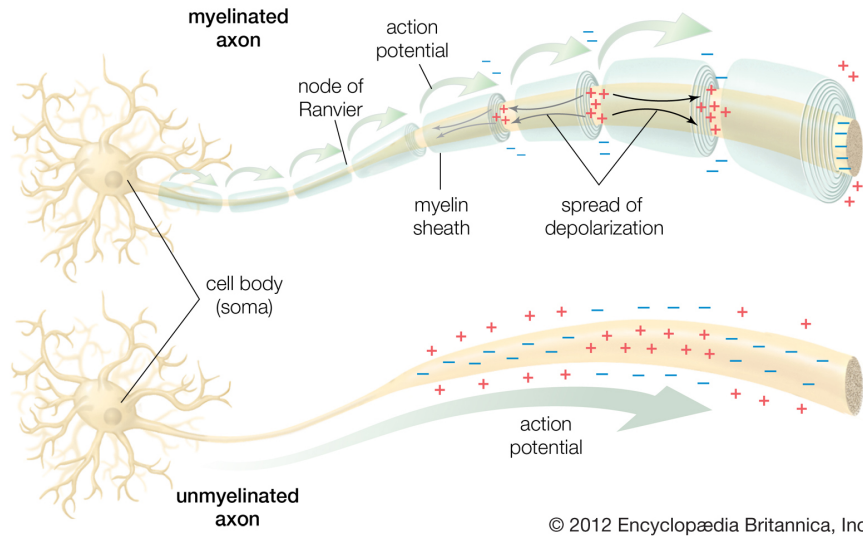
feedback mechanism. The potential arising in one point of the membrane excites the adjacent parts that are still at rest. This causes further depolarization and the potential propagates. In theory this would happen in both directions, but given that the newly activated areas have a refractory period in which they lose their excitability characteristic, along the axon this happens in a unidirectional way.

As already mentioned, the action potential usually propagates along the axon in the direction that goes from the dendrites of the current neuron to the synaptic terminations of the axon: this propagation modality is called *orthodromic*. In particular conditions, such as in the case of artificial stimulation of the axons, the action potential can also travel in the opposite direction: this phenomenon is called *antidromic* propagation.

### Myelinated fibres

In myelinated fibres, the axon is surrounded by Schwann cells. Each Schwann cell wraps around a longitudinal segment of the axon of 0.5 to 1 mm [3] in a spiral conformation, producing so-called *lamellae*. These cells form the myelin sheath. Between the segments of myelin there are small unmyelinated regions, called Nodes of Ranvier (NOR). Ions cannot flow freely through myelin, but they can through NORs. Consequently, APs can arise only at the node level, and are conducted from node to node, in a *saltatory conduction*. The current is transmitted through

the extracellular liquid and through the axoplasm (that is the cytoplasm of the axon) from node to node, exciting the successive nodes one after the other, with a considerable energy saving and a 5 to 50-fold increase in the velocity of propagation.



**Figure 2.4:** Comparison between myelinated and unmyelinated fibres

### Conduction velocity

The conduction velocity (CV) of an impulse along the axon is an important neurophysiological parameter. It depends on the distance that the suprathreshold depolarization covers along the active zone. The parameters on which this distance depends are the diameter of the axon (at which increase the longitudinal resistance of the axoplasm decreases) and the transverse resistance of the outer covering membrane. In myelinated fibers, thanks to the high transverse resistance, it is possible to reach speeds of 100 m/s with axonal diameters of 20  $\mu\text{m}$ .

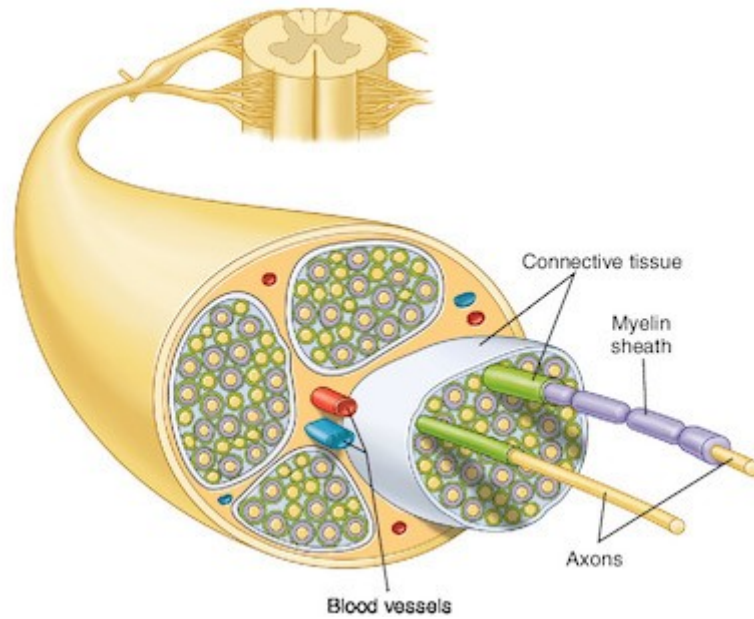
In mammals, peripheral nerves contain myelinated fibers with a diameter between 0.5 and 20  $\mu\text{m}$  and a conduction velocity between 3 and 100 m/s, and unmyelinated fibers of less than 2  $\mu\text{m}$  with a velocity of 0.5 to 2 m/s[3]. The fiber groups were classified into three groups (A, B, C) with decreasing conduction velocities, both for the afferent and efferent part. Afferent fibers also have a second, more recent classification in groups I, II, III, IV. An overview of the correlation between diameter, CV and function is provided in table 2.1.

Diameter and Conduction Velocity			
	<i>Diameter (<math>\mu\text{m}</math>)</i>	<i>CV (m/s)</i>	<i>Example</i>
Myelinated	15-20	60-100	Skeletal muscle
	5-15	30-90	Vibration, touch
	1-10	6-60	Muscle spindle
	1-5	6-30	Deep pressure
Unmyelinated	0.5-2	0.5-2	Pain, tickle

**Table 2.1:** Fibres in the PNS and related conduction velocities, adapted from

### 2.1.3 Peripheral Nerves

The main function of the peripheral nerves is to rapidly conduct electrical signals between the central nervous system and the target organ. The primary structural component is the peripheral nerve axon. A peripheral nerve structure includes thousands of axons guaranteeing their structural and functional support.



**Figure 2.5:** Structure of a peripheral nerve

The axons, within the nerve, are grouped into fascicles, held together by a sheath of connective tissue called the *epineurium*, aimed at providing protection against traction and compression and nourishes the fibers through a vascular network. The fascicles are delimited and enclosed by a membrane, the *perineurium*. Inside the



perineurium, the axons are surrounded by fibroblasts, Schwann cells and capillary endothelium cells, each enveloped by an additional connective tissue membrane, the *endoneurium*. The nerve contains both afferent and efferent fibers. The former have a sensitive function, the latter motor. The cell bodies of the neurons that compose the nerve are generally grouped in or near the CNS, where nerves originate.

From their origin, peripheral nerves depart as bundles, which then branch distally to reach the target organs. The topographical distribution of the fibres undergoes changes from the proximal to the distal section accordingly, as evidenced in the peripheral nerve characterization studies presented in section 3.2.1.

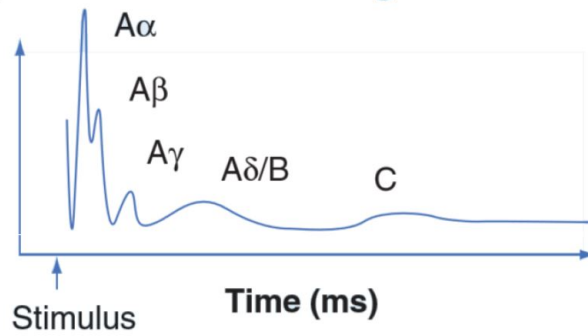
## 2.2 Introduction to Peripheral Nerve Interfaces

Since the discovery of the excitability of nerve tissues, the efforts of many researchers have focused on studying the nervous system through electrical interaction with it. The studies involved the two possible aspects, namely stimulation and recording, leading to the development of peripheral nerve interfaces, some of which are now commercial, based on the use of electrodes that stimulate or record nervous activity. The excitability properties of the tissue allow to intervene on it, stimulating it by inducing the motion of charges and evoking its response. Several methods for peripheral nerve stimulation are studied and some are also applied in bioelectronic devices now widely used in clinical practice for the treatment of some pathological conditions. Chronic recording does not have the same success as stimulation, and the aim of the following sections is to introduce the main problems encountered in pursuing this goal and some opportunities for overcoming them.

### 2.2.1 The electroneurogram

The electrical signal obtained from the peripheral nerve is called electroneurogram (ENG). This signal strongly depends, in its characteristics, on the recording method. ENG recordings can be performed mainly in two conditions, of natural or electrically evoked signals. For natural conditions, here, are also intended those that are induced by mechanical stimulation, causing a response through the somatic sensory pathways. The electrically evoked ones, as the name suggests, are instead obtained by subjecting the nervous structure to electrical stimulation, through more or less invasive electrodes. In the case of electrical stimulation of the nerve and not of the single fiber, the response of the system gives rise to a waveform called Compound Action Potential (CAP), constituted by the superposition of the action potentials of the recruited fibers that propagate in the nerve. These potentials travel at different conduction velocities, therefore the CAP has a different shape depending on the position with respect to the stimulation electrodes in which it is recorded. The

fibers, when electrically stimulated, are recruited in an order that depends on their diameter: the thickest fibers are recruited first. The ENG signal recorded *in vivo*, in particular the one of natural origin, is difficult to read due to its energetic and spectral characteristics. In fact, it shows a bandwidth between  $0.5 - 7\text{kHz}$ , with main energy in the range  $0.5 - 3\text{kHz}$  and amplitude between 1 and  $300\text{ }\mu\text{V}$ , that is highly dependant on the recording method and experimental conditions. The signal is immersed in noise, especially electromyographic signal from surrounding muscles, that has an overlap in spectrum and is in the order of millivolts in amplitude. This implies that particular attention in the design of the electrode and the processing units must be paid, in order to exclude these noise sources as much as possible, maximising the possibilities of interpreting the signals correctly.



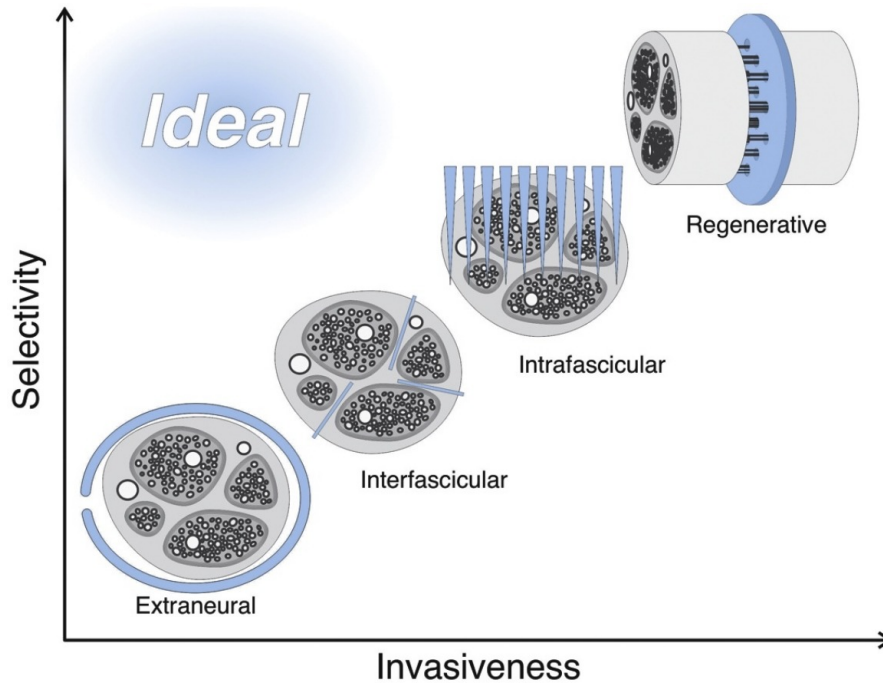
**Figure 2.6:** Shape of a compound action potential (CAP)

### 2.2.2 Electrodes for Peripheral Nerves

Electrodes are the first interface between the nervous system and a bioelectronic device. The basic principle concerning electrodes, both for recording and stimulation, is that the closer the electrode is to the target, the higher will be its selectivity [2]. Several electrode typologies have been developed over the past decades, used both in research and commercial devices. According to the aforementioned principle, they are characterised by different degrees of selectivity, and consequently invasiveness, related to implant stability and durability. It is necessary to underline, in fact, that the increase in selectivity and invasiveness causes greater damage to the nerve structures, and consequently can lead over time to a greater risk of implant failure. In each type of implant, after the wound healing process for restoring the homeostasis lost due to the surgical procedure, the continuous presence of the implant contributes to overstimulate the immune system. This leads to chronic inflammation and to implant encapsulation, determining in some cases implant stabilisation, but in some others implant failure or reduction of functionality [6], especially in the case of chemical and mechanical irritation, where it can result

in extreme inflammation, edema, hemorrhage or necrosis [2]. Keeping into account such an important aspect is the fundamental prerequisite for the development of a device that is chronically implantable on humans. The most invasive electrodes, due to their characteristics, are those that have revealed the best performances for achieving selectivity, however they are also those that have the shortest duration due to the destructive impact they have on the structure of the nerve.

**Electrode classification** The electrodes for interfacing with the PNS are first of all divided into intraneural and extraneural. This classification is based on the need for the electrode to pierce the outer membrane of the nerve, the epineurium. More in detail, the main categories in which existing electrodes can be divided are extraneural, interfascicular, intrafascicular and regenerative. Their placement into a selectivity-invasiveness plane is shown in figure 2.7. Existing technologies are positioned along a straight line in which, as said earlier, selectivity increases step by step with invasiveness. The ideal condition, for which high selectivity can be achieved with low invasiveness, cannot be reached with electrodes alone, but it is possible to approach it by means of some precautions in terms of contact configuration and processing algorithms. These categories can be described as



**Figure 2.7:** Selectivity and invasiveness of the existing peripheral nerve electrode categories, compared with the ideal case. Image from [2].

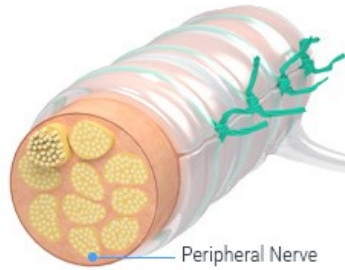
follows:

- **Extraneural electrodes:** they are electrodes that surround the epineurium, avoiding to compromise the structure by penetrating in it, and thereby minimizing damage to the tissue. They are the least invasive category, but as a disadvantage they allow a minimal and not very selective interaction with the fascicles [7]. The most common example, the cuff electrode, is crucial in this work and further discussed in the next sections.
- **Interfascicular electrodes:** they penetrate the epineurium, but not the BNB and the other structures. [2]. Interfascicular electrodes combine the simplicity of extraneural electrodes with the closer contact to axons and the greater stimulation selectivity of intrafascicular electrodes. The electrodes occupy the space between the fascicles allowing the central axons to be better recorded and stimulated [8]. Both afferent and efferent fibres are in reach of these types of electrodes, achieving a more selective interfacing than the previous modality [7]. However, even if axonal damage was not visible after the implant, chronic applications have not been reported [9].
- **Intrafascicular electrodes:** they penetrate the protective membranes of the nerve, allowing for better selectivity and increasing the SNR of Navarro recordings. These structures utilize penetrating electrodes which are pushed into the nerve bundle, directly through the fascicles. Electrodes in this category have all demonstrated reliable interfacing with the PNS, making selectivity possible by acquiring signals from a small number of axons [7].
- **Regenerative electrodes:** they differ from the others in function. In fact, unlike the other categories, they are not implanted on an intact portion of the nerve. They consist of sieve-like structures implanted on and a residual transected peripheral nerve [7] with the aim of rejoining the two terminations. They are designed to interface high numbers of fibers, placing the electrodes around the holes in which the resected fibers can grow back. Axons reconnect through the sieve, allowing both stimulation and recording of signals [2].

For choosing among these kind of electrodes for long-term applications, it is important to which extent the electrode can be used chronically *in vivo*. To this purpose, it must be considered that chronic physiological or histological damage must be minimised, in order to assure a long-term performance. Inflammatory reaction should influence recording or stimulation as little as possible, but high spatial resolution should be achieved in most cases, which often but not necessarily implies an increase in the number of electrodes, that is not always desirable [10].

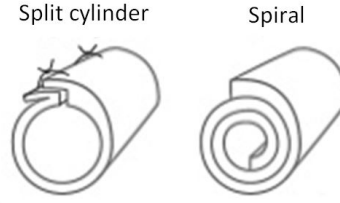
### 2.2.3 Cuff electrodes

Cuff electrodes are the only kind of electrodes whose use on human for about a decade has been reported. In fact, Christie et al. showed that spiral cuff electrodes for stimulation can be stable and working for up to 10.4 years on peripheral nerves [11]. They were among the first to be investigated for use with implanted neural devices and have been extensively applied in both research applications and commercial neural devices for the last four decades [9]. They consist of a tubular structure made from an insulating material, like silicone rubber or PDMS, with metal electrodes, commonly made of platinum, placed on its internal face. The tube is positioned around the nerve, as shown in figure 2.8. Despite the proven *in vivo*



**Figure 2.8:** Example of a cuff electrode

durability of this electrode type, the nerve can still be damaged, especially in the limbs where range of motion is wide. Therefore, electrodes should be flexible and possibly self-sizing [9], and should guarantee an easy, fast and repeatable surgical procedure at the same time. During implantation, the cuff is opened, inserted and closed. The cuff should guarantee a close contact between the electrodes and the neural tissue, minimizing the body liquid in-between which may cause effect of less efficient electrical stimulation and reduced recording amplitudes due to a current shorting effect. There are numerous mechanisms that allow to fasten the cuff around the nerve. The most used ones are the split cylinder and the spiral cuff electrodes. The first are tubes with a cut in the longitudinal direction, that makes them C-shaped in section, and requires to be closed by suturing its adjacent sides after being placed on site. The latter are capable of automatically wrapping around the nerve, thanks to the fabrication techniques used, and allow faster placement, self-sizing, and better response to nerve swelling after implantation, with less risk of nerve damage.



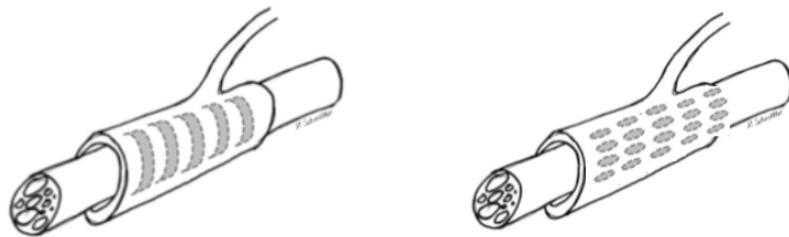
**Figure 2.9:** C-shaped and Spiral cuff electrode designs

### Recording with cuff electrodes

Cuff electrodes can be used both for stimulation and recording. In recording, the cuff electrode brings with itself the need to use strategies aimed at improving the discrimination of different sensed activities. To enhance recording performances, the strategies concern the manufacture of the cuff, the number and arrangement of contacts, the pre-processing strategies and those of interpretation using algorithms. Long and snug-fitting cuffs generally provide larger signals [12], whereas improved reproducibility in the fabrication process avoids noise pickup due to mismatch of contact impedances [9].

Moreover, cuff designs have evolved from single ring cuffs to more complex models, including a high number of contacts for collecting signals at different positions on the nerve surface increasing the number of contacts from which to extract information. The most effective and used strategies to date are indicated in this work as *multielectrode cuff* (MEC) and *multicontact cuff* (MCC). The MECs are cuff electrodes that have a high number of rings, usually more than three. MCCs are similar, but the rings are split into multiple contacts which constitute different channels for the amplifiers. These two designs are shown in figure 2.10.

MECs and MCCs are the basis from which it is possible to start to achieve selectivity in extraneural recordings. Their coupling with appropriate processing strategies enables a better understanding of the information traveling along the nerve.



**Figure 2.10:** Multielectrode cuff (left) and multicontact cuff (right) designs.  
Credits: M. Schuettler

## Chapter 3

# State of the Art

### 3.1 Selective recording strategies

As it has been extensively discussed in chapter 2, cuff electrodes have low selectivity. They only sense a general, multiunitary activity, as the action potentials from individual axons summate, not allowing the identification of single spikes and featuring considerable spatial and temporal averaging [13]. In addition to not being functionally specific to a particular fiber, the recordings of these signals are dominated by large and fast nerve fibers and often with low signal-to-noise ratio (SNR). As a result, strategies are needed to achieve sufficient granularity [14].

In the cuff, the first strategy is to increase the number of contacts from which to extract information. Relatively large numbers of contacts provide multiple recording sites that can be used to determine conduction velocity [12] or improve spatial selectivity. In this regard, cuffs are developed having a highly varied number of contacts, like in the case of MECs and MCCs.

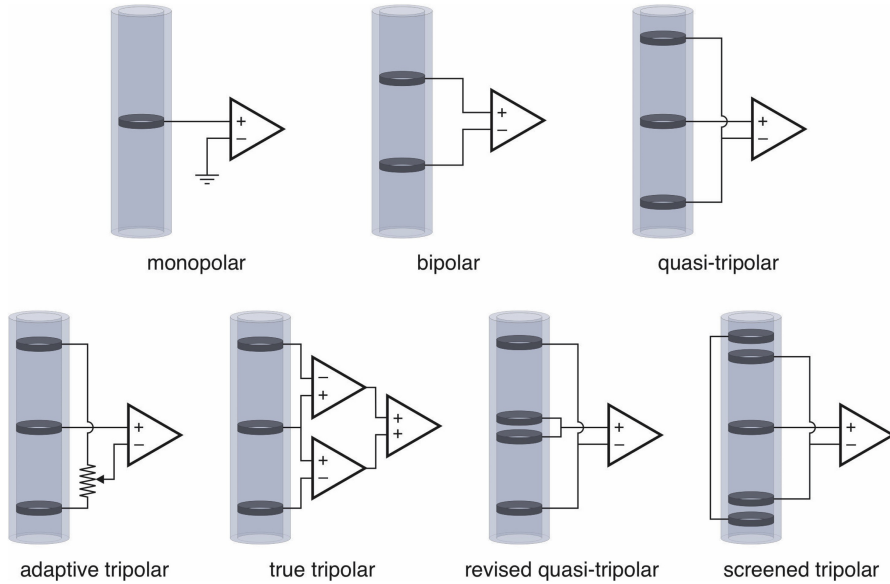
#### 3.1.1 Improving selectivity with the processing hardware

Several research groups have studied the *in vivo* recording of mechanically and electrically evoked activity with cuff electrodes, both with MEC (i.e. [15, 16, 14] and MCC (i.e. [17, 18, 19]) coupled with processing techniques to achieve selectivity.

As anticipated in section 2.2.1, *in vivo* ENG recordings are affected by noise, especially EMG. For its rejection, it has been largely proven in the literature that an efficient method for improving quality of the recorded signals, excluding distant noise sources, is to use tripolar configurations in the first stage amplifiers of the recording circuit. This configuration is able to capture the signal's major features with some EMG contamination [2]. On contrary, in the monopolar and bipolar recordings during *in vivo* experiments, the neural signal is dominated by EMG,



that makes it visible just in case of denervation [2]. Tripolar configuration benefits from the linearisation of the potentials coming from external sources. As the cuff has an insulating effect, potentials can only flow inside from the outer edges of the cylinder traveling from one side to the other and creating a linear gradient. The tripolar configuration, creates a reference using the two adjacent electrodes to the recording electrode and creating a reference electrode which is virtually in the same position as the recording electrode, allowing to subtract external noise to itself and giving just the neural signal as an output. Obviously, this occurs only in ideal conditions, without taking into account impedance imbalances and assuming that the electrodes are perfectly equidistant. To compensate for non-idealities, tripolar configurations are developed in different variants, shown in figure 3.1 that can include shielding or correction in impedance differences. In addition to these



**Figure 3.1:** Tripolar configuration and its variants [2]

strategies, analog and digital filtering can help remove the remaining noise, whereas pattern recognition and signal processing can provide useful information on the type of signals and the amount of elicited nerve activity [13].

### 3.1.2 Improving selectivity with processing algorithms

To finally achieve selectivity and being able to discriminate the kind of signals that are being conveyed through the nerve, classification algorithms are commonly applied and often optimised on the type of cuff electrode used.

### Multielectrode cuff strategies

The MECs provide longitudinal spatial information, being made up of numerous rings arranged equidistant along the nerve. Consequently, they can give a clear view of the variation of the waveform along the nerve course. Since fibers inside the fascicles have different diameters and therefore different CVs for their action potentials, the recorded waveforms differ from point to point. This is particularly evident in the morphology of CAPs, described in the 2 section. This feature is exploited by the so-called Velocity Selective Recording (VSR) algorithms, investigated in several works ([20, 21, 22]), described here in the hypothesis in which the signal is collected with electrodes organized in a tripolar configuration. VSR algorithms demonstrated the ability to discriminate signals on the basis of propagation direction (afferent or efferent) and conduction velocity. These algorithms assume that, for each fiber, there is a delay between the appearance of the signal between two successive outputs of the tripoles, that is a function of the spacing between the tripoles, constant, and the CV, that has different values depending on the type of fiber. Although CV can be calculated even using only a single pair of electrodes in bipolar configuration, it has been shown that the velocity selectivity of such a system can be increased by using multiple electrodes, helping to better discriminate between fiber populations with adjacent CVs.

Algorithms used for VSR are *delay-and-add* VSR plus a variant coupled with Artificial Neural Networks. In the *delay-and-add*, the signal of each channel is delayed with respect to the last one by an interval depending on the distance of the channel from the last one and the speed that is being searched. A delay  $\tau$  is established for each velocity. In the case of N electrodes, the N-1 electrode is delayed by  $\tau$ , N-2 by  $2\tau$  and so on. The signals thus obtained from each channel are summed together. In doing so, the signal components generated by fibers with similar speed to the one associated  $\tau$  with tau add up to each other. The resulting signal is bandpass filtered, with frequency centered on the CV sought, and the resulting signal peak allows to quantify the number of fibers with that specific CV. The procedure is repeated for various values of  $\tau$ . For each velocity,  $\tau$  will be given by:

$$\tau = \frac{d}{v}$$

with d distances between the electrodes and v signal conduction velocity. As it can be imagined, these methods have better performance at low speeds, for which higher resolution can be achieved.

### Multicontact cuff strategies

The MCC combines, to the longitudinal selectivity of the MECs also a transversal selectivity, due to the fact that the rings are divided into small contacts that touch

the nerve in several points along the circumference, without mediating the voltage at different points with the same longitudinal position. This additional information leaves room for the development of algorithms that try to locate the active fibers within the nerve or that recognize a particular activation pathway. The main approaches can be divided into two categories, based on what has just been said: Source Localization algorithms and Spacetime Information based algorithms.

**Source Localization** Source localization algorithms consist of algorithms that try to reconstruct the position of the internal sources that produced the resulting signals outside the nerve by solving an inverse problem. To solve it, it is necessary to know the solution of the forward problem, encoded in a matrix called the *Leadfield*. In the main approach [23], this matrix constructed starting from a FEM model. However, the difficulty in creating appropriate modeling of the nerve has led to investigating methods of solving the inverse problem without the need for a model [24]. The solution of the inverse problem must be carried out for every instant of time, resulting in a high computational cost.

**Spacetime Information** This method is based on recording the signals resulting from the activation of specific pathways using an MCC, which is able to obtain space-time information. These signals, generally CAPs obtained by electrical stimulation, are then used to train classifiers and use them to discriminate new signals. Each pathway has ideally associated an  $M \times N$  matrix, where  $M$  and  $N$  are respectively the number of rings in the MCC and the number of contacts per ring, highly specific and different from that associated with a different pathway. These matrices are reshaped into vectors, exploited to create tailored matched filters [25] or as input for classifiers such as Artificial Neural Network or Random forest [26].

**Approaches for other types of electrodes** Other approaches exist that are applied to other kinds of electrodes, especially Flat Interface Nerve Electrode (FINE). They can be divided mainly into two categories, Spatial filters and Blind Source Separation algorithms. The former include Bayesian Spatial Filters [27] and Beamforming approach [28], while the major example of the latter is Independent Component Analysis [29]. It is not excluded that also these methods may be applied to cuff electrodes, even if with appropriate modifications.

### 3.1.3 The need for simulations

In general, based on the experience gained, it emerges that having electrodes with a high number of contacts increases the possibilities of discriminating the different components of the ENG signal recorded by extraneural electrodes. Having a higher number of electrodes, small and with uniform impedance, would be the ideal

List of the main classifiers for ENG extraneural recordings	
<b>Multielectrode cuff (MEC)</b>	
Velocity selective recording (VSR)	Delay-and-add VSR Delay and add VSR + ANN
<b>Multicontact cuff (MCC)</b>	
Source localization	FEM leadfield Experimental leadfield
Spatiotemporal information	Matching filter ANN or Random forest
<b>Others</b>	
	Bayesian classifiers Beamforming Blind source separation Independent component analysis

**Table 3.1:** Summary of the most popular processing algorithms for selective recording

solution. However, this is complex to achieve due to the cost and the technological feasibility of manufacturing. In addition, the greater size of the processing units (in particular due to the increase in recording channels and the number of amplifiers), the consequent increase in consumption of the acquisition system, and the greater number of data to be transferred and processed must be taken into account. A compromise is therefore necessary. In order to investigate possibilities without wasting too much time and resources, *in silico* simulations offer an attractive opportunity.

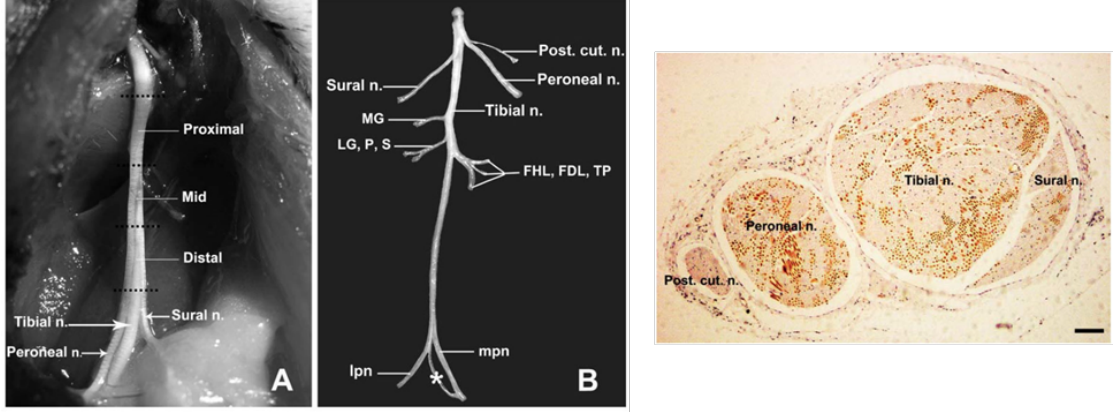
## 3.2 Peripheral Nerve simulations

### 3.2.1 Nerve characterisations

In order to have a simulation approach, in addition to knowing the generic structure of a nerve, it is necessary to understand the studies conducted for its characterization, regarding topography, composition and electrical characteristics. These anatomical studies are used to understand the assumptions to be made in the case of a simulation and provide a yardstick for evaluating the obtained results. Studies conducted on the sciatic nerve of mammals, in particular the rat, are a benchmark, as many *in vivo* tests on PNIs are conducted on these animals.

The rat sciatic nerve the nerve is composed of 4 fascicles, which run together in the

proximal part and separate distally. The nerve first splits in two, forming the cutaneous branch and the peroneal nerve on one side, the tibial and the sural nerves on the other. The first fascicle to split as an independent one is the cutaneous branch, followed by the peroneal nerve. Finally the tibial and sural branches separate. The anatomy is shown in figure 3.2.



**Figure 3.2:** Anatomy of a rat sciatic nerve [30].

H. Schmalbruch [31] made an analysis of the fibre composition in the rat sciatic nerve. His anatomical study has been conducted in order to understand the fiber composition of these branches. In order to distinguish the motor and the sympathetic fibres, de-efferentiation and sympathectomy were performed. The measurements were performed by counting and by computations on light and electron micrographs. The reported number of myelinated fibres was 7800, of which 4500 for tibial nerve, 1900 for the peroneal nerve, 1050 for the sural nerve, 350 in the cutaneous branch. The number of unmyelinated fibres was around 19.400. The partition of fibres is reported in table 3.2.

Axon count in rat sciatic nerve				
	<i>Tibial n.</i>	<i>Peroneal n.</i>	<i>Sural n.</i>	<i>Cutaneous br.</i>
<b>Efferent</b>				
Myelinated motor	1000	600	-	-
Unmyelinated sympathetic	3700	1100	1500	100 <sup>2</sup>
<b>Afferent</b>				
Myelinated	3500	1300	1100	400
Unmyelinated	5400	3000	2800	1800

**Table 3.2:** Fiber counts in the rat sciatic nerve, adapted from [31]

Badia et Al. [30] have instead studied the topographic distributions of the different

types of fibers within the different fascicles, and their study can be taken into consideration for whole-nerve simulations with special focus on location of the active fascicles. In general, as stated in their work, topographical knowledge will enable to plan the design and implantation of electrodes in adequate positions on the nerve, facilitating selective recording.

As for the electrical properties, different models of the nervous tissue have been developed at different levels. Starting from the properties of axons, the most popular and used models for simulation studies, which exploit the electrical characterization of the constituent elements of the axon (reported in table 3.3) are the Hodgkin-Huxley [32] and the MRG axon [33] models. These ones are the basis of the most common higher-scale simulations.

Axon components				
<i>Parameter</i>	<i>Fixed</i>	<i>Max.</i>	<i>Min.</i>	<i>Units</i>
<b>Unmyelinated</b>				
Membrane conductivity		0.0008	0.000144	S/cm <sup>2</sup>
Membrane specific capacitance	0.81			μF/cm <sup>2</sup>
<b>Myelinated</b>				
Membrane conductivity in the regions				
<i>Node of Ranvier</i>		1.95	0.018	S/cm <sup>2</sup>
<i>Paranode</i>	0.001			S/cm <sup>2</sup>
<i>Juxtaparanode</i>		0.0082	0.002	S/cm <sup>2</sup>
<i>Internode</i>		0.00269	0.002	S/cm <sup>2</sup>
Membrane specific capacitance	2			μF/cm <sup>2</sup>
Myelin specific capacitance per lamella	0.05			μF/cm <sup>2</sup>
Myelin conductivity per lamella	0.0005			S/cm <sup>2</sup>

**Table 3.3:** Adapted from [34].

To set up a simulation it is necessary to be aware of the properties of the extracellular medium, the membranes, and in general the tissues that surround the nerve fiber inside the nerve, understanding which strategy would be the best one for modeling, in the future, the overall activity of the nerve, and anyway for having a comparison on which to evaluate the performance of a model.

In table 3.4, the main electrical properties of the extracellular fluids in the nerve are summarized.

As reported by J. Hope et Al. [34], the impedances of the passive tissues like perineurium, epineurium and endoneurium obtained from mammalian peripheral nerves have never been reported. Commonly, the used value for perineurium (47,600 Ω.cm) is the one from frog, reported by Weerasuriya et Al. [35], whereas the one for epineurium (1,211 Ω.cm) is usually taken from the measurement, by Ranck et. Al.

Fluid and extracellular volumes		
<i>Parameter</i>	<i>Fixed value</i>	<i>Units</i>
Intracellular fluid resistivity	70	$\Omega\text{.cm}$
Periaxonal fluid resistivity	70	$\Omega\text{.cm}$
Extracellular space resistivity	1000	$\Omega\text{.cm}$

**Table 3.4:** Electrical properties of fluids and extracellular volumes in the nerve. Adapted from [34].

[36], of the transverse impedance of the dorsal column of a cat. For extracellular resistivity instead, values range between 175 and 1,000  $\Omega\text{.cm}$ . Moreover, both Hope [34] and Aristovich et Al. [37] underlined that the overall electrical impedance of the nerve changes when transmembrane potential of the axons changes, due to the opening of their ion channels.

### 3.2.2 Overview of the simulation approaches

The simulation approaches for the nerves are varied and, as can be imagined, depend specifically on the application for which they are developed. It starts with models that simulate individual nerve fibers, up to models of entire nerves. The approaches are developed for stimulation and recording, although those for stimulation, based on the research done, are much more numerous than those for recording, especially at the scale investigated in this work. Of the tools developed, some are open-source while others, in greater number, are not made available by the research groups that have developed them, while still others are commercial software. In this section an overview of the existing simulation approaches will be provided. These are presented according to a bottom-up criterion, therefore starting from single-fibre models up to whole-nerve simulations.

### 3.2.3 Axon models

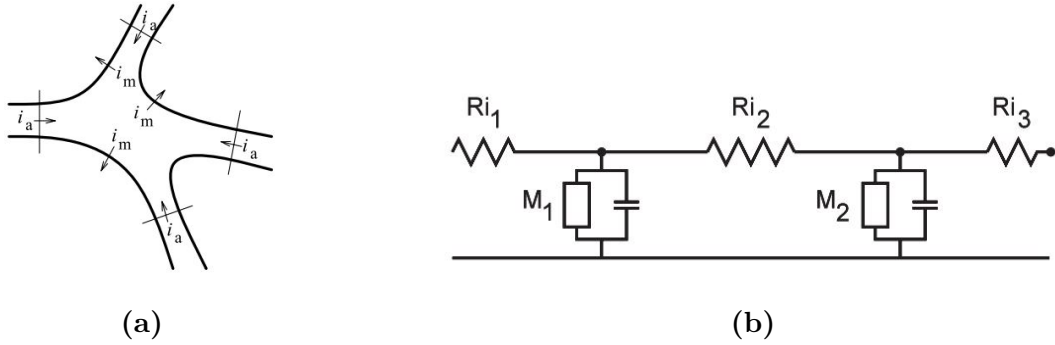
The models for the individual axons are developed in order to simulate ion fluxes and the generation of action potentials within the nerve cells, and in most cases include the application of *cable theory* to neuron compartmental models. According to this approach, a neuron can be represented as a cylinder divided into sections connected to each other and to the external environment through a network of resistances and capacities. The equation that generically describes the relationship between current and voltage is

$$\frac{\partial V}{\partial t} + I(V, t) = \frac{\partial^2 V}{\partial x^2}$$

which undergoes a spatial discretization, giving rise to a set of differential equations, to be solved for each compartment into which the neuron is divided, of the type

$$c_j \frac{dv_j}{dt} + i_{ion_j} = \sum_k \frac{v_k - v_j}{r_{jk}}$$

which represent an application of the Kirchhoff's current law [38]. As shown in figure 3.3a, the current entering each compartment through the adjacent ones, that is the right term in the equation, must be equal to the current flowing through the membrane in its capacitive and ionic component, on the left. The schematization of the nerve is therefore that represented in figure 3.3b, in which the components on top represent the axonal compartments, with their own internal resistance, connected through resistances and membrane capacities to the extracellular space.



**Figure 3.3:** Currents in a compartmental model of a neuron (3.3a) and electric representation of a neuron (3.3b) according to the model. From [38].

The membrane models have evolved from the use of simple resistors to active models with voltage dependent properties [39].

One of the first examples of this type of model dates back to the late 80's and is that of Sweeney et Al. [40], who created a compartmental model consisting of NORs and internodal space, the first having voltage-gated sodium channels, passive loss channels and nodal capacitance, the latter as a perfect insulator due to the presence of myelin.

Although in some cases similar models have been developed in calculation software (e.g. Matlab), the software of choice for such modeling, to date, is NEURON [38]. This tool was developed by Hines and Carnevale, from Yale University, and provides a powerful and flexible tool for simulating neurons and creating networks with them, in order to study their electrochemical signals. Applications are more frequent for the brain, since it is a software that allows the creation of networks, however, being programmable by the user, it is well suited for the creation of *ad hoc* models, highly customisable. Many research groups create their own models



and make them available on the ModelDB platform [41]. NEURON is based on the application of cable theory but allows the user to bypass mathematical modeling and to deal just with concepts belonging to neuroscience.

Several models are developed in NEURON and used for peripheral nerve simulations. One is the Frankenhaeuser-Huxley model [42], which takes up the Squid Giant Axon Model by Hodgkin-Huxley [32] and adapts it to the results on the toad. This is a “space clamped” model, in which the propagation of the stimulus along the fiber is not simulated [39]. Another, widely used and based on mammalian experimental data, is MRG axon, by McIntyre et Al. [33]. It was developed to model the excitability and recovery cycle of myelinated axons, and is capable of reproducing what was observed in the experimental case, reproducing the membrane dynamics. A double-cable structure is used to model myelin, Nodes of Ranvier, and adjacent transition structures. This model, already used by itself, is widely exploited as a building-block for higher-scale simulations.

### 3.2.4 Extracellular medium simulation

Scaling up, the next approach includes the modeling of the extracellular medium in which the axons are immersed, moving towards a reading of the extracellular potential that also takes into account geometric aspects.

A preliminary study on the characteristics of the extracellular medium was conducted by Grill [43] who developed a computational model to determine - under the hypothesis of a purely resistive medium - the effects of isotropy and anisotropy on the extracellular excitation of the fibers.

#### Single fibres in idealized isotropic medium

An example of this implementation - applied to spinal cord - is by P. Sacre et Al. [44]. This method doesn’t reflect the fact that, in the real case, a nerve includes hundreds of fibers ensheathed as a bundle. Another example of implementation of this method, implemented in neuron, is by Parasuram et Al. [45], and is available at NEURON database ModelDB [41].

#### Electrically anisotropic interstitial space

These models are based on the observation that currents in a nerve flow more freely in longitudinal direction rather than in transversal direction. An example of this implementation, although it is for stimulation, is by Grill [43]. An example for the determination and modeling of nerve resistivities in different directions is provided by Sacre [44].

### 3.2.5 Simulations with multiple axons

More complex approaches take possibly into account the fact that in the nerve there are multiple axons, with different diameters, and/or the topographic distribution of fascicles inside the nerve, possibly including the presence of layers of membranes dividing the fascicles.

#### Bundle model

An example of this approach has been developed by Hayami et Al. [46] for simulating nerve conduction, and was optimized for the simulation of the recording of CAPs. Nerve is modeled as a bundle of nerve fibers, each fiber modeled as a cable of NORs. The single action potentials were simulated for each fiber, and then a CAP was calculated by using weighted sums in order to take into account the contribution of all the fibers. This model still does not take into account the geometric characteristics of the nerve, and furthermore the results on conduction velocities partly do not fully reflect the experimental data.

### 3.2.6 FEM-based approaches

FEM-based approaches have the advantage of introducing explicitly the dimensionalities of the structures, allowing to include electrode geometries and topographic information about the nerve into the model.

Most of these methods have been developed for stimulation simulations. They generally couple a FEM model in which to calculate the electric fields, to be then applied to a cable model in NEURON or Matlab.

An example of this approach is that of A. Rapeaux et Al. [47] which simulates in COMSOL the components of extracellular fields perpendicular to the fibers and uses them within the NEURON software, applying them to the Frankenhauser-Huxley model for the study of selective stimulation of nerve fibers. Another example is by Pelot et Al. [48]. Here the distribution of potentials is calculated by COMSOL and imported into NEURON as well, to investigate the effect of modeling the perineurium and endoneurium on simulations for stimulating peripheral nerves. The 2D and 3D models developed try to include the electrical parameters of these sheaths and the geometric parameters of the bundles, approximating them, to quantify which parameters affect the stimulation thresholds the most, also evaluating the influence of two different designs of cuff electrodes.

A totally FEM-based approach was developed by D. Holder research group and discussed in several papers [37, 49, 39]. This approach responds to the need of having a fully bi-directional coupled model of the nerve fibers with the extracellular spaces.

The research group has, in fact, developed the model in order to study Electrical Impedance Tomography, an innovative technique that allows to discriminate the activation of fibers in the nerve on the basis of impedance variations that occur during the propagation of APs [37].

The technique consists in applying alternating currents using extraneural electrodes, reading the transverse impedances, and then applying an algorithm (whose principle is similar to the FBP algorithm used in Computed Tomography) that allows to reconstruct the positions of the activated fascicles and consequently the type of information that is traveling on the nerve.

The need for bidirectional communication led to the creation of space clamped 1D FEM models of the fibers and their integration within 2D and 3D nerve models in COMSOL, with the support of Matlab [39]. The model is complex, its bases start from models already examined, but the purpose for which it was developed and for which it is optimized is different from the objective of this thesis and, moreover, there is still no existing multi-fiber model, as only simulations with single fibres have been developed to date. The fibres used as building-blocks were developed both for the myelinated case [39], which includes an obviously non-uniform structure involving the double-cable paradigm, and the unmyelinated one [49].

### 3.2.7 Whole-nerve models

General-purpose models that try to provide a very flexible support are needed for investigating selective recording. For this objective, it must be taken into account the fact that recording in the PNS is deeply affected by the method used, that shapes the nature of the extracellular space. In recent years, software has been implemented with the objective of providing a more general model of PNs.

Among these, one of the most notable examples is Sim4Life, developed by Zurich MedTech. It consists of a combination of human phantoms and tissue models, and offers the possibility of multiphysics simulations optimized for biological problems. It is an extremely powerful tool, but it is commercial software.

C. Lubba et Al. [50] have instead developed an open-source toolbox, PyPNS, proposed as an alternative to commercial software such as Sim4Life. PyPNS was developed in Python and in conjunction with NEURON, for simulating a bundle of many axons, and use the NEURON simulator to model axon membrane processes, using myelinated and unmyelinated axons in the diameter range found in the periphery.

It exploits the quasistatic approximation of Maxwell equations - that makes the fields separable in time and space - and simplified nerve geometry to improve efficiency avoiding to run FEM simulations repeatedly, using precomputed fields. This comes to the strong assumption of circular symmetry of the extracellular medium.

The intended use of the tool is to describe one peripheral nerve that consists of an arbitrary number of both myelinated and unmyelinated fibers. It is possible to choose among different types of fiber activation mechanisms, extracellular media and recording mechanisms. When a simulation is run, the definition of each axon is sequentially transmitted in NEURON, together with the associated excitation mechanism. After calculating the membrane processes in NEURON, the extracellular single fiber action potential (SFAP) is calculated in PyPNS, and when every fiber is processed the compound action potential (CAP) is finally computed.

# Chapter 4

## Tools and Method

Based on what has been analyzed in the State of the Art, no simulation environment is available that allows the generation of electroneurograms controlling the histological and topological characteristics of the nerve and recording them through models of cuff electrodes whose design can be defined according to the user's needs. Hence the need to create a highly customizable environment that can satisfy these requirements.

The simulation approach presented here has been developed by continuously investigating possibilities offered by the software, making refinements with the objective of creating a reliable simulation environment that can be used for flexibly testing and optimising the selective recording from peripheral nerves using cuff electrodes. The solution developed relies on three environments, with different functions: NEURON, COMSOL and MATLAB. The first two are used to produce data at two different levels, as explained in the next sections, whereas the last one is used for processing the results and creating plots.

### 4.1 Overview of the model

The objective of the model is to simulate neuronal activity, which produces currents that propagate inside the nerve, generating variations in potential, and to read these from the outside, using cuff electrodes.

The fundamental elements of the model are, consequently, the axons of peripheral neurons, which constitute the nerve fibres and work as current sources, and the volume of the nerve itself, in which the fibres are dispersed. Outside the nerve, the cuff electrode is placed. Nerve and cuff are surrounded by an environment which consists of the animal's tissues, in the case of *in vivo* tests, or of saline solutions, in the case of *ex vivo* tests.

The construction of the model began with the simulation of the individual axons within the NEURON software. It was chosen to start with the simulation of myelinated fibers only. From these simulations the membrane currents at the nodes of Ranvier have been exported.

Correspondingly, in COMSOL, the FEM model of the nerve was created, including the generation of the geometry of the nerve and of the axons contained in it. For simplicity, the axons have been modeled as one-dimensional structures, parallel to the axis of the nerve, consisting of the succession of points representing the NORs, arranged at regular intervals according to the simulation parameters in NEURON. Current sources have been assigned to these points, reproducing currents waveforms simulated in NEURON and imported into COMSOL.

In this model, the structure of the cuff electrode is added. Its structure is modeled so as to be able to vary the geometric parameters and test different configurations of the metal contacts.

After calculating the solution, the voltage values on the contacts can be exported (for example in MATLAB) to allow processing according to the user's needs.

## 4.2 Simulation of the axons

### 4.2.1 The NEURON Simulation Environment

Axon modeling was done in NEURON simulation software [38]. As anticipated in chapter 3, this software allows to apply on cable theory applied to compartmental models of neurons for studying their neurophysiology. The software allows to generate neuronal models in which the neural cell is divided into segments, representing soma and neurites, which are in turn divided into sections. For each section, the associated differential equations are defined and solved by applying numerical methods of integration that guarantee computational efficiency and stability of the solutions.

The models are defined by means of an object-oriented syntax. NEURON incorporates two programming languages, one based on `hoc` (High-Order Calculator, derived from C language) and the other one on Python, among which it is possible to choose. In this thesis, programming in `hoc` has been exploited. Through the programming tools it is possible to insert the properties of the sections and the various simulation conditions. In particular, a standard sequence includes the creation of the model topology, the assignment of anatomical and biophysical properties, the insertion of the stimulation electrodes and the control of the [38] stimulation waveform. The program also includes a graphical interface that allows to control the simulations and to represent the results. The resulting environment is extremely flexible, and does not necessarily require the user to have direct knowledge of the

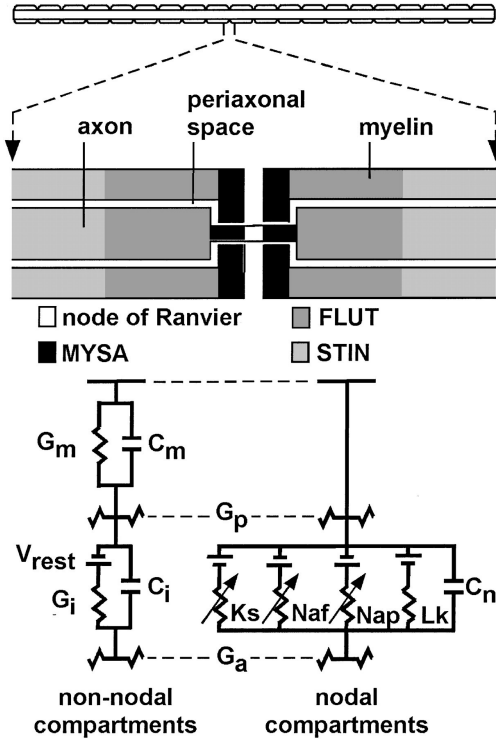
mathematical bases necessary for setting up a compartmental model, allowing to deal only with the neurophysiological entities of neurons, even if it is possible also to create new mathematical models of the membrane dynamics.

### 4.2.2 Modeling the axons

For addressing the problem of modeling axons, a pre-built model has been exploited, like in many works presented in section 3.2.2. In theory, accurate modeling of the nerve would require modeling all types of fibers that are present in it, both myelinated and unmyelinated (as done in [50], which constitutes an excellent reference point in this work) and also taking into account also their diameter distribution. Having to start from a model with few axons, the simulation of only myelinated axons, which are faster and which give a greater contribution to the output signal due to the greater intensity of the action potentials produced by them, was preferred. This approach is common in the literature, where a widely used model is the MRG axon [33]. The MRG model is an excellent representation of myelinated fibers in the mammalian peripheral nerve, widely exploited as a building block in several peripheral nerve simulations, including FEM models. It is particularly accurate thanks to an explicit representation of the morphology of the fiber. Unlike other models, where the axons are simply represented as a succession of NORs and myelinated sections, here the internodal segments are more detailed, including the myelin attachment segment (MYSA), the paranode main segment (FLUT), and the internode segment (STIN), represented in figure 4.1. The resulting fibre consists of a periodic repetition of the node-MYSA-FLUT-6\*[STIN]-FLUT-MYSA-node sequence. Membrane dynamics includes linear and non-linear models based on experimental observations, developed by the research group and inserted into NEURON.

MRG axon basic script allows to choose among 9 different possible diameters of myelinated fibres. In its original version, downloadable from ModelDB [41], it allows to create an axon made of 21 Nodes of Ranvier. The model initialises the components and sets a stimulus, injecting current in the central NOR.

The original script has been modified in order to fit the needs of the overall model, mainly in two ways. First, a simulation needs an almost continuous distribution of diameter values in order to be realistic. Therefore, an extrapolation approach has been set up in order to get the values for the non existing diameters, and the script has been adapted. This approach is similar to the one used in PyPNS [50]. Given the need of a series of first order approximations when passing into COMSOL, however, the extrapolation model has not been used in the final setup, as the number of axons inserted in the COMSOL model is limited, and the need for having a distribution of diameters (that would be needed in case of a realistic number of axons inserted) is not present yet.



**Figure 4.1:** Graphic representation of compartments in the MRG axon model [33].

Given the limited number of diameters, an automation of the creation (like a loop creation) of several axons on which running automatically simulations, was not considered a priority, but must be considered as a future development in the case of extension to a high number of fibres and a continuous distribution of fibre diameters.

Second, in order to fit the needs of the simulation, some parameters needed to be changed, and the scripts have been modified accordingly. The first modification concerns the number of NOR and thereby the total length of the axon. In fact, the MRG axon model sets the number of NOR, not the overall length of the axon, that depends on the dimension of the sections, shown in table 4.1, and on their number, that is originally fixed as reported in table 4.2. In this case, however, the length is the crucial parameter and, in the COMSOL model, the it will be declared in order to build the FEM model of the nerve. Therefore, priority must be given to this aspect. Given the limited number of diameters for which the operation needed to be performed, even if also this procedure could be automated, MRG axon scripts for each diameter were modified manually. An example of the final code is presented in the Appendix (A.1.2).



The number of nodes of Ranvier for a given diameter was computed using Excel. The results are shown in table 4.3.

Default geometrical parameters in MRG axon - 21 nodes					
<i>Fibre d.</i> <i>FiberD</i> ( $\mu\text{m}$ )	<i>Node l.</i> <i>nodelength</i> ( $\mu\text{m}$ )	<i>MYSA l.</i> <i>paralength1</i> ( $\mu\text{m}$ )	<i>FLUT l.</i> <i>paralength2</i> ( $\mu\text{m}$ )	<i>Internodal l.</i> <i>deltax</i> ( $\mu\text{m}$ )	<i>Computed</i> <i>total l.</i> (mm)
5.7	1	3	35	500	10
7.3	1	3	38	750	15
8.7	1	3	40	1000	20
10	1	3	46	1150	23
11.5	1	3	50	1250	25
12.8	1	3	54	1350	27
14	1	3	56	1400	28
15	1	3	58	1450	29
16	1	3	60	1500	30

**Table 4.1:** Original settings for MRG axon

Topological parameters			
<i>Entity</i>	<i>Variable name</i>	<i>Original value</i>	<i>New definition</i>
NORs n.	axonnodes	21	(user-defined, table 4.3)
MYSA n.	paranodes1	40	$2 * (\text{axonnodes} - 1)$
FLUT n.	paranodes2	40	$2 * (\text{axonnodes} - 1)$
STIN n.	axoninter	120	$6 * (\text{axonnodes} - 1)$
Total n. of sections	axontotal	221	$\text{axonnodes} + \text{paranodes1} + \text{paranodes2} + \text{axoninter}$

**Table 4.2:** Redefinition of the original topological parameters as a function of the computed node number.

So, instead of using the fixed values proposed in the original MRG Axon script, the number of NOR has been redefined and the number of the other sections has been parametrized as its function, as visible in table 4.2.

Furthermore, according to the observation – that is in agreement with what happens in the real case – that when changing the axon diameter it's necessary to change the value of the amplitude in the *IClamp*, the stimulation current has been changed from the original value of 2 nA to 5 nA, in order to have the maximal response also on the largest fibres (16  $\mu\text{m}$ ).

Finally, as the internodal length changes according to the fiber diameter, for simplicity and for increasing realism of the simulation, the current is now injected in

Computed parameters for simulations of a 6cm nerve fibre		
<i>Fibre diameter</i> ( $\mu m$ )	<i>Number of</i> <i>internodal segments</i>	<i>Number of</i> <i>NORs</i>
5.7	120	121
7.3	80	81
8.7	60	61
10	52.17	53
11.5	48	49
12.8	44.44	45
14	42.86	43
15	41.38	42
16	40	41

**Table 4.3:** List of the axon geometrical parameters computed for a 6 cm nerve fibre

the first node of each axon, in order to have the same starting point for current injection in the FEM for each axon.

The execution of the MRG axon code is followed by the execution of other scripts created *ad hoc* to export the right parameters needed for the FEM model. The sequence is executed within the script *mosinit.hoc*, that contains inside, among the others, the following:

- *MRGaxon.hoc*
- *vec\_generationLoop.hoc*
- *currentConversion.hoc*
- *fileCreationLoop.hoc*

The code corresponding to *mosinit.hoc* and the functions inside it are reported in the Appendix, section A.1.

*vec\_generationLoop.hoc* is a script that executes the steps necessary to save the membrane currents as a vector. The parameters that are exported are the vector. *tvec* with the timesteps of the simulation, and one *i\_membrane* vector for each NoR node section, containing the value of the membrane current density over the NoR surface over time (in the timesteps saved in *tvec*).

*I\_membrane* is an available distributed mechanism in NEURON, is an object property for each section and its value is available at `node.i_membrane()`.

Distributed mechanism means that the property in question is normalised on the surface it's calculated on. In this case,  $I\_membrane$  is expressed in  $\text{mA}/\text{cm}^2$ .

*currentConversion.hoc* is a script that converts each current  $i\_membrane$  in a point process. As an  $i\_membrane$  equivalent as a point process seemed to be unavailable, an *ad hoc* conversion script had to be created, since the NOR in COMSOL will be modeled as point current sources, needing a value of current expressed in Amperes. Therefore, current conversion takes the value of diameter of the fiber and length of the node of Ranvier, calculates the surface of the NOR, and multiplies it for the current density in  $I\_membrane$ , also applying, in the code, a conversion factor that is necessary to transform the current value from  $\text{mA}/\text{cm}^2$  to A, making it ready to be used into COMSOL. The equation applied for conversion is:

$$I_{PCS} = I_{mem} * S_{NOR} = I_{mem} * 2\pi r_{axon} l_{NOR}$$

where  $I_{PCS}$  is the current for the equivalent point current source,  $I_{mem}$  is the membrane current density,  $S_{NOR}$  is the NOR surface, that takes into account the axon radius  $r_{axon}$  and the NOR length  $l_{NOR}$ .

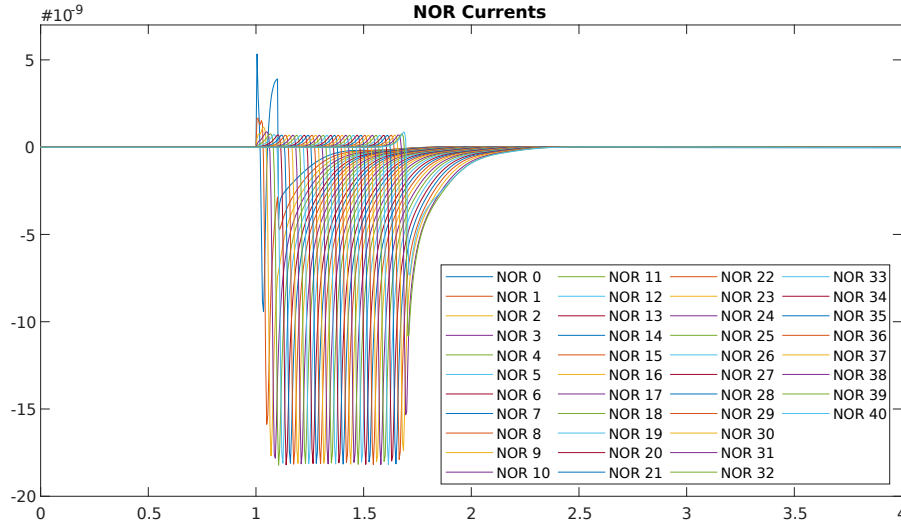
*fileCretionLoop.hoc* finally creates the files. The chosen format is *.txt*, formatted as a table in which the first column contains the time steps and the second column contains the value of the converted membrane current. In the file, the path for the creation of the files is specified, and a file for each node is created, changing the name of the file accordingly. The procedure is completely automated in order to make it possible to deal with large amounts of data. Currents are computed for each possible diameter. To quickly verify the process worked, currents are imported and plotted in Matlab in a superposed graph.

The Matlab script that allows to produce this graph is *NORcurrent\_plot.m* (see Appendix, A.3.1. An example of output graph for this kind of verification is shown in figure 4.2. The delay between the waveforms is a proof of the right propagation of the action potential. This is how the currents look like, before being imported into COMSOL.

In saving membrane currents, the following assumption has been done: as myelin sheath is modeled, in the MRG axon, as a highly resistive medium, the current passing through it can be considered negligible, in first approximation. Therefore, to streamline the process and increase computational efficiency, just the currents in the nodes of Ranvier have been exported, assuming that in the real case they are the ones that contribute most to the potential read by an extraneural electrode.

### 4.3 Multiphysics Model for PN Recording

The model used for nerve recording has been developed using the software COMSOL Multiphysics, that allows to deal with the geometrical modeling of the structures,



**Figure 4.2:** Currents for all the nodes of Ranvier in a 16 micrometers axon

adding material properties but also custom functions for defining what physically happens to them.

### 4.3.1 Modeling the system with COMSOL Multiphysics

COMSOL is a very popular simulation environment for studying engineering problems. It consists of a cross-platform that contains finite element analysis and multiphysics simulation. The software allows to model components by defining their geometry, materials and physics to be applied. The components created are divided into elements by creating a mesh, and after this it is possible to perform various types of studies, for which the software proceeds with numerical solution of partial differential equations defined on each element through an iterative process. The results are saved and made available in multiple formats for their visualisation and their subsequent processing. The various functions of the software are described in the documents available at [51], and are briefly described, where necessary for an explanation of the choices made, in the next paragraphs.

The final model includes mainly 3 entities: the nerve, the cuff, and the saline solution in which both are immersed. This choice was made both to be able to have a match with ex-vivo tests, and for avoiding to introduce complexity for modeling biological tissue, that in most cases features also the intrinsic electrical activity of the muscles surrounding the nerve.

To simplify the modeling process, the cuff geometry, which is the most complex to define, is created in separate files and then imported into the nerve model file. In

the following, the modeling methods are presented, with an overview of the most relevant tasks necessary to reach the final objective.

### **4.3.2 Cuff electrode geometric model**

A cuff electrode can be defined as a tube made with an insulator material that contains metal contacts connected with wires. There may be different kinds of closing mechanisms that allow to place the cuff around the nerve, as described in chapter 2. The purpose of this study is to provide a tool to evaluate the use MEC and MCC electrodes.

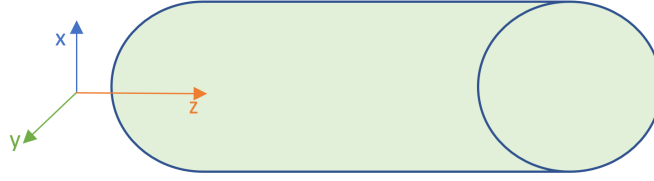
As already explained in chapter 2, the MEC consists of several metal rings placed along the tube, whereas in the MCC the rings are splitted into several contacts along the circumference. Within the same category, the contacts can differ in number, distance, surface area. The COMSOL model has been set up in the way that these parameters can be changed effortlessly.

#### **Assumptions**

The cuff electrode has been modelled as a tube with metal contacts, neglecting the closing mechanism and the wires inside the cuff structure, as the primary objective is the one of reading the potentials on the epineurium and see how the different configurations of the metal contacts affect the interpretation of the signal. What matters most, therefore, is the outer surface of the metal contact that is close to the nerve and immersed in the fluid. The goal is to read the potential on the contacts, all the elements that have a neglectable influence on the result are not modelled, such as the necessary connection wires with the amplification chain, assumed to be extremely thin and connected with an infinite impedance.

#### **Approach**

The model has been set up considering the need of easily modifying configurations with less effort as possible. Therefore, the model can be defined parametrised, as several parameters have been inserted to facilitate the definition of the geometrical properties. Two different geometrical models have been created, one for multicontact cuff electrode and one for multielectrode. In fact, since the MCC requires more adjustments and definitions than the MEC, the two models have been kept separated and each one is optimised for the specific purpose. Anyway, the approach is the same and most parameters are shared among the two. In creating the model, the reference frame has been set as shown in figure 4.3. Considering the cuff as a cylinder, its longitudinal direction coincides with the  $z$  direction of the reference frame. The same axes are used also as a reference for the nerve structure.



**Figure 4.3:** Reference frame for the cuff and the nerve models

## Parameters

Some parameters in the geometry definition are independent and user-defined, others are dependant and are calculated based on the user's inputs. This allows to rapidly change geometry, without recomputing all the parameters each time. The choice of the user-defined parameters has been done in the sense of making the geometry definition as intuitive as possible. As an example, talking about the cuff, it has been decided to specify inner diameter (that is in close relationship with nerve diameter) and thickness of the insulator, whereas the outer diameter of the tube is calculated accordingly. This choice has been preferred to the one, considered less intuitive, of specifying inner and outer diameter. The parameters used in the model, grouped as described below, are listed in table 4.4. The list of the dependant parameters, together with the expression used for their calculation from the user-defined ones, is presented in table 4.5.

**Position in space** This group includes the parameters that allow to specify in which position with respect to the reference frame the cuff electrode must be created. It's not meaningful for the geometry itself, but fundamental when coupling the electrode with the nerve model.

**Cuff geometry** The parameters of cuff geometry are the same for both MEC and MCC and they concern the properties of the insulating tube enclosing the nerve and containing the electrodes.

**Electrode geometry and Electrode configuration** These set of parameters concern the electrode contacts, in particular their dimension and their configuration, intended as the position on the cuff structure.

<b>Cuff electrode model parameters</b>		
<i>Name</i>	<i>Model</i>	<i>Description</i>
<b>Position in space</b>		
<i>z_pos</i>	MEC, MCC	Position of the electrode in the reference frame along the longitudinal (z) axis
<b>Cuff geometry</b>		
<i>Cuff_length</i>	MEC, MCC	Longitudinal dimension of the cuff
<i>Cuff_thickness</i>	MEC, MCC	Thickness of the cuff (difference between outer and inner radii)
<i>Cuff_r_in</i>	MEC, MCC	Internal diameter of the cuff
<i>Cuff_r_out</i>	MEC, MCC	External diameter of the cuff
<b>Electrode geometry</b>		
<i>N_electrodes</i>	MEC, MCC	Number of rings in the cuff
<i>N_contacts</i>	MCC	(Must be less or equal to 12)
<i>El_thickness</i>	MEC, MCC	Radial Thickness of the metal contact
<i>El_jut</i>	MEC, MCC	Projection of the metal contact from the inner wall of the cuff
<i>El_Height</i>	MEC, MCC	Height (longitudinal thickness) of each ring contact
<i>El_Radius_in</i>	MEC, MCC	Inner electrode radius
<i>El_Radius_out</i>	MEC, MCC	Outer electrode radius
<b>Electrode configuration</b>		
<i>Edge_dist</i>	MEC, MCC	Distance of the outer rings from the edges of the cuff structure
<i>El_spacing</i>	MEC, MCC	Distance between the electrodes
<i>Cont_ratio</i>	MCC	Ratio of contact over total circumference
<i>Cont_dim</i>	MCC	Dimension (in degrees) of the single contact

**Table 4.4:** Table of the geometrical parameters used in COMSOL for the definition of the cuff electrode

## Functions

The function definition allows to create flexible ways for controlling the variables values in the creation of a model. For the cuff geometry creation, two functions have been created: the first is *Elpos*, that defines the longitudinal position of electrode x, while the second is *Contpos*, that defines the angle with respect to the x axis at which the contact x in a MCC ring must be positioned. These functions are

Dependant parameters in cuff geometry definition	
<i>Parameter</i>	<i>Expression</i>
Cuff_r_out	Cuff_r_in+Cuff_thickness
El_Radius_in	Cuff_r_in-El_jut
El_Radius_out	El_Radius_in+El_thickness
El_spacing	(Cuff_length-2*Edge_dist)/(N_electrodes-1)
Cont_dim	360/N_contacts*Cont_ratio

**Table 4.5:** List of the dependant parameters and the expressions for calculating their value starting from user-defined ones

declared as shown in table 4.6.

Functions for cuff geometry creation	
<i>Function name</i>	<i>Expression</i>
Elpos(x)	z_pos+Edge_dist-El_Height/2+El_spacing*(x-1)
Contpos(x)	360/N_contacts*(x)

**Table 4.6:** Functions for the calculation of the contacts positions along the z axis and the circumference, respectively.

### Creation of the geometry

Geometry is built using the COMSOL tools for creating geometries. Other approaches could have been to create the geometry by means of an external software, like AutoCAD or Solidworks. The cuff is created in COMSOL using solid modeling. During solid modeling, a geometry is formed as a combination of solid objects using Boolean operations like union, intersection, and difference. Such entities are known as composite solid objects. Dimension used is 3D, to have similar development processes both for MEC and MCC, even if for the multielectrode cuff it could have been possible to use 2D axisymmetric. The cuff is basically a cylindrical structure, therefore it's created by generating cylinders and applying Boolean operations to them. In the case of the MCC, the steps are the same of the MEC, plus some additional operations.

**Creation of MEC model** The creation of the geometry for the multielectrode model follows a sequence of creation of cylinders, boolean operations and an array operation. First, the tube for the insulator material is created, using the difference of two cylinders: *Out\_cuff* with radius *Cuff\_r\_out* and height *Cuff\_length* at position (0,0,*z\_pos*), and *In\_cuff*, identical but with radius *Cuff\_r\_in*.



Then, the same is done to create the first electrode ring. The creation of the electrode structure consists of the creation of cylinder *Out\_electrode* with electrode radius *El\_Radius\_in* and height *El\_Height*, at position  $(0,0,Elpos(1))$ , where *Elpos(x)* is the position along the longitudinal (*z*) direction of the ring *x*; the creation of *In\_electrode* as above, but with radius *El\_Radius\_out*; and a difference operation between the two.

Finally, an array of electrodes is created, through a replication of the electrode structure as an array with *x* and *y* size equal to 1, and *z* size equal to the number of rings. Displacement is equal to *El\_spacing* in the *z* direction (0 in the others). This command creates a *x-y-z*-size duplication of the selected entities along the chosen directions, at the chosen spacing.

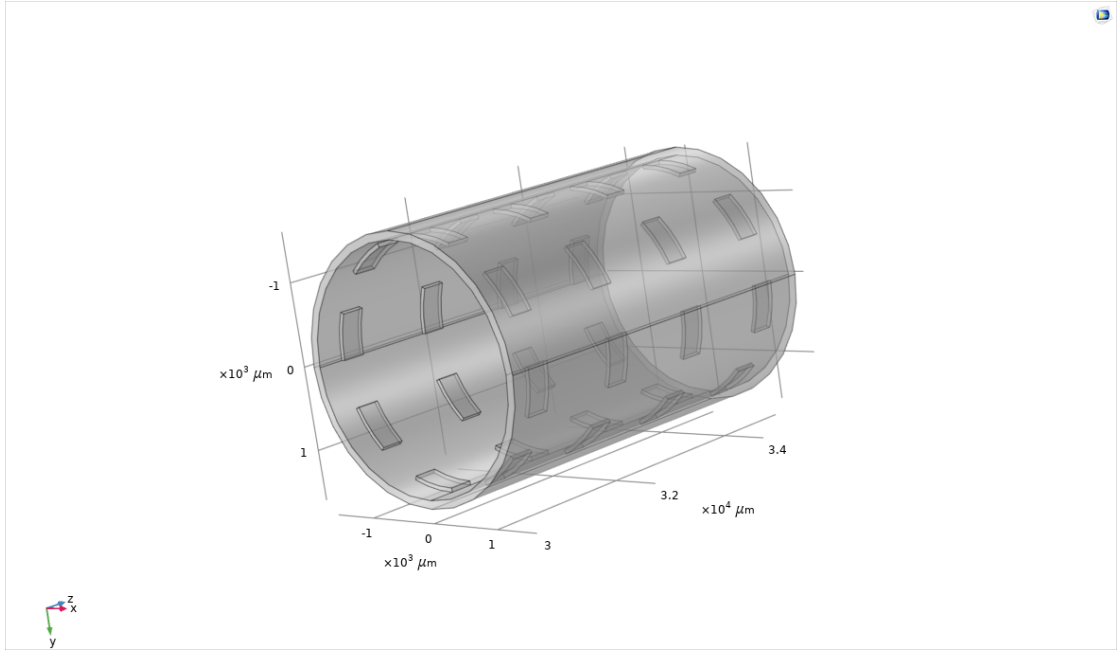
**Additional steps for MCC model** For creating MCC, domain partition, revolution, and logic operations are exploited. First, the ring structure of the electrode is partitioned in a half, allowing it to show its section.

One of the halves is deleted, and the section surface is revolved of an appropriate angle that goes from 0 to *Cont\_dim*. Also the remaining half is deleted, leaving just the revolved element. Then, a series of nested *ifs* is exploited (resulting in the equivalent of a for loop, that doesn't exist among the options), in order to replicate as many contacts as the user specified in *N\_contacts* along all the circumference, checking at every step if the number of contacts requested is upper than the current number of created contacts. Until the final number of contacts *N\_contacts* is reached, the sequence keeps adding a new contact by applying a rigid transform (rotation) of an angle defined with the appropriate function, *Contpos(x)*, where *x* is the number of the contact that is being created. The procedure has a maximum number of iterations, equal to 12, but this upper limit can be easily increased just by adding other nested *ifs* in the sequence.

An example of MCC, resulting from the operations that have been described above, is shown in figure 4.4.

### 4.3.3 Saline medium and nerve segment model

Nerve is geometrically modeled as a cylinder, containing axons inside, parallel to its axis. As outlined previously, the basic assumption is that axons are considered as straight lines containing equally spaced points, that are the nodes of Ranvier. The nerve is immersed in a saline medium, that will be responsible of the electrical coupling with the cuff electrode. Also the saline medium is modelled as a cylinder, with a diameter that is a function of the nerve diameter. Parameters for the nerve structure are listed in table 4.7 and are declared in micrometers. The axons inside the nerve are modelled according to the corresponding parameters used in



**Figure 4.4:** Example of a MCC created in COMSOL.

Nerve structure parameters	
<i>Parameter</i>	<i>Description</i>
Nerve_length	Height of the cylinder modeling the nerve
Nerve_diameter	Diameter of the cylinder modeling the nerve
Saline_radius	Radius of the cylinder modeling the saline solution

**Table 4.7:** Parameters used for the definition of the nerve structure and the saline medium

NEURON, that have been imported into COMSOL for each fibre as tables. The complete set of fibre parameter tables created into COMSOL has been reported in table 4.8.

The crucial parameters that need to be imported are the diameter, the node number, and the internodal length. Nerve model creation includes the execution of repetitive tasks, both for the creation of an extremely high number of nodes and the insertion of currents, that is extremely burdensome and time consuming even for a very small number of axon fibres. This would imply the unsustainability of the design task, if not automated. Therefore, after proof of concept about the feasibility of operations, the need for automation of these tasks has been solved using the Application Builder in COMSOL.

Neuron parameter sets imported into COMSOL		
<i>Label</i>	<i>Parameters</i>	Value
1. Fibre 5.7	Diam_1 ( $\mu\text{m}$ )	5.7
	Interlength_1 ( $\mu\text{m}$ )	500
	Node_number_1	121
2. Fibre 7.3	Diam_2 ( $\mu\text{m}$ )	7.3
	Interlength_2 ( $\mu\text{m}$ )	750
	Node_number_2	81
3. Fibre 8.7	Diam_3 ( $\mu\text{m}$ )	8.7
	Interlength_3	1000
	Node_number_3	61
4. Fibre 10	Diam_4 ( $\mu\text{m}$ )	10
	Interlength_4 ( $\mu\text{m}$ )	1150
	Node_number_4	53
5. Fibre 11.5	Diam_5 ( $\mu\text{m}$ )	11.5
	Interlength_5 ( $\mu\text{m}$ )	1250
	Node_number_5	49
6. Fibre 12.8	Diam_6 ( $\mu\text{m}$ )	12.8
	Interlength_6 ( $\mu\text{m}$ )	1350
	Node_number_6	45
7. Fibre 14	Diam_7 ( $\mu\text{m}$ )	14
	Interlength_7 ( $\mu\text{m}$ )	1400
	Node_number_7	43
8. Fibre 15	Diam_8 ( $\mu\text{m}$ )	15
	Interlength_8 ( $\mu\text{m}$ )	1450
	Node_number_8	42
9. Fibre 16	Diam_9 ( $\mu\text{m}$ )	16
	Interlength_9 ( $\mu\text{m}$ )	1500
	Node_number_9	41

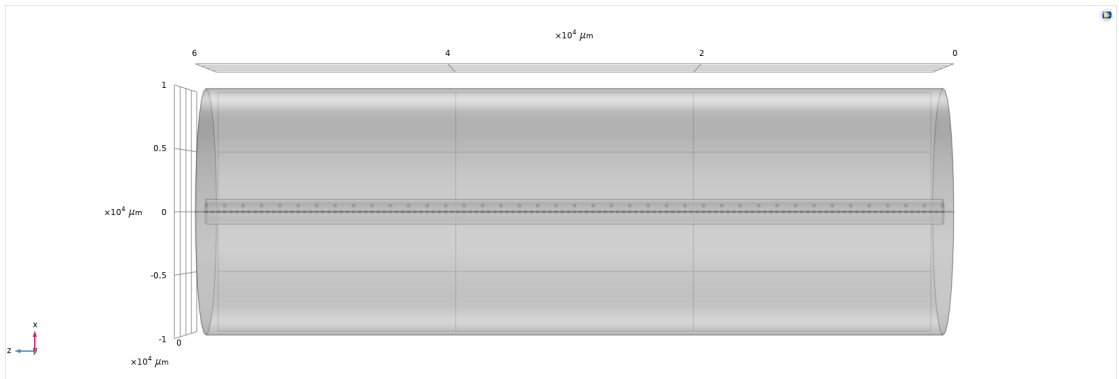
**Table 4.8:** List of the NEURON parameters imported in COMSOL

### Nerve geometry creation

Creation of the nerve geometry is composed mainly of two tasks, with different grades of complexity: the creation of the nerve and saline domains, and the creation of the axons and the NORs. For the creation of nerve and saline medium, the operations are the simple creation of two cylindrical structures. In particular, for the nerve segment a cylinder with radius  $Nerve\_radius$  and length  $Nerve\_length$  is created, for saline, instead, the cylinder has the same length as the nerve but radius  $a*Nerve\_radius$ , with  $a$  arbitrarily defined in each simulation, but generally

equal to 10.

For the creation of the axons, the diameter and location of the axon, defined with the transversal coordinates ( $x$  and  $y$ ), are declared in the parameters for the new axon to be created. The new axon parameters are added to a specific list, to keep trace of all the axons that have been inserted in the model. Once parameters have been set, the axon is created by inserting points at the transversal location ( $x,y$ ) along the longitudinal direction  $z$ , with spacing defined as the internodal length for that specific fiber diameter, and listed together with the previously created axons. The first nodes, for all the axons inserted, are at the specified transversal location and at position 0 along the longitudinal axis. This means that the first NOR, that is also the one where the stimulating current is injected, are all in the same location, and that the potential will propagate along the longitudinal direction, showing itself gradually at increasingly greater  $z$  coordinates. The points are therefore inserted and will work as current sources further in the simulation. An example of the set of nerve, saline and axon geometries is shown in figure 4.5.



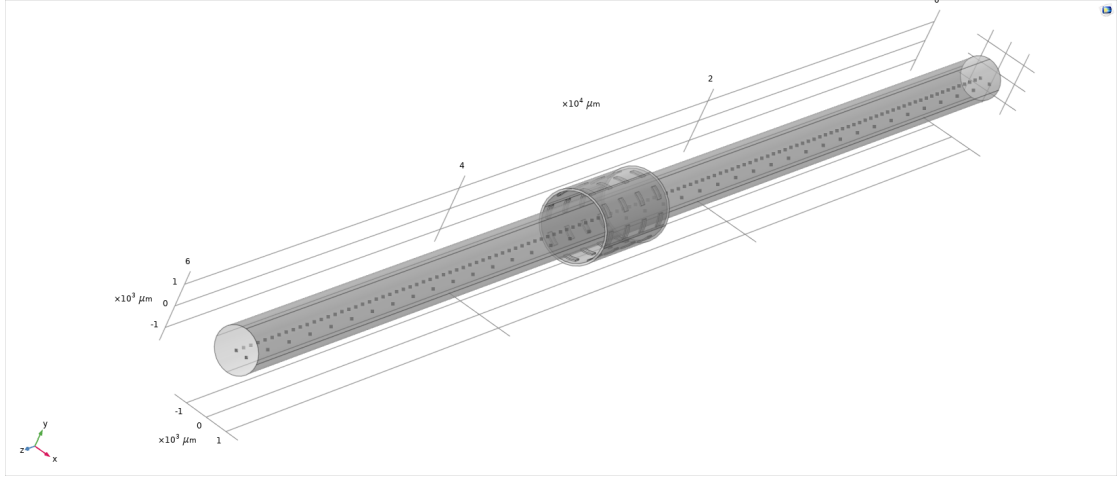
**Figure 4.5:** Nerve, saline and axon geometries, in the case of a 5.7 and 16  $\mu\text{m}$  diameter axons. It is possible to notice the difference in spacing between the NORs.

### Automation of geometry creation using the application builder

As anticipated, the creation of the axons is a burdensome task. The time it would take to do it manually would make the final objective of creating a complex structure practically unfeasible. COMSOL Multiphysics offers the application builder, that is a tool for creating GUIs that can also be used to integrate programming tools into the COMSOL workflow. Therefore, specific scripts in Java programming language have been written to automate the most time consuming tasks, to be run in specific parts of the workflow for creating the geometries.

### Coupling nerve and cuff models

At this point in the workflow, the procedure is complete for coupling the electrode geometry with the nerve geometry and proceed with setting the other features of the model. An example of the coupling between the nerve and the cuff structure is presented in figure 4.6.



**Figure 4.6:** Nerve and cuff geometries, coupled together. The cylinder representing the saline solution is hidden.

#### 4.3.4 Setting the materials

The materials inserted in the model are 4 in total, chosen among COMSOL material library or user-defined. For building a realistic model we need to assign, at least: an insulating material for the cuff structure; a metal for the electrode contacts; a saline solution for the conductive medium between the nerve and the electrode; a model of the nervous tissue material. In order to conduct electric currents simulation with the AC/DC module, two properties are always required by the software: Electrical conductivity  $\sigma$  and Relative permittivity  $\epsilon_r$ .

##### Cuff structure

For the cuff structure, PDMS was chosen. This material is available in the materials library in COMSOL and is commonly used in the creation of cuff electrodes extensively in research and biomedical industry. PDMS is an insulator, therefore, the material defined in the COMSOL material library misses the conductivity value,

that has been set to  $1/(4\text{e}13)$  S/m. Values for PDMS electrical properties are shown in table 4.9.

PDMS			
<i>Property</i>	<i>Variable</i>	<i>Value</i>	<i>Unit</i>
Electrical conductivity	$\sigma$	$1/(4\text{e}13)$	S/m
Relative permittivity	$\epsilon_r$	2.75	-

**Table 4.9:** PDMS electrical properties

### Metal contacts

For metal contacts, platinum was chosen. Also this material is commonly used in the fabrication of cuff electrodes, and is present in the COMSOL material library. Platinum is a conductor. The value missing, in this case, was the relative permittivity. The research conducted for finding this value has led to extremely dispersed and discordant values, and the number of sources found was insufficient to have a reliable value. However, some considerations have been done, like comparing the values found for platinum with values for similar other metals, that have led to temporarily choose a value of 7, that will need refinement in the future. The values of the electrical properties for platinum are shown in table 4.10.

Platinum			
<i>Property</i>	<i>Variable</i>	<i>Value</i>	<i>Unit</i>
Electrical conductivity	$\sigma$	8.9e6	S/m
Relative permittivity	$\epsilon_r$	7	-

**Table 4.10:** Platinum electrical properties

### Saline solution

The saline solution is completely user defined, as the material didn't exist in the COMSOL's library. The saline solution chosen is NaCl 0.9%, commonly used in *ex vivo* experiments. Reported values for saline conductivity span in literature from 14 mS/cm to 18 mS/cm for 0.9% NaCl, whereas relative permittivity is in the order of 1 [52]. The chosen values are in the table 4.11.

### Nerve tissue

Unlike the previous materials, nerve tissue requires a finer definition because propagation in this medium is the most crucial aspect influencing the potential that

Saline			
<i>Property</i>	<i>Variable</i>	<i>Value</i>	<i>Unit</i>
Electrical conductivity	$\sigma$	1.45	S/m
Relative permittivity	$\epsilon_r$	1	-

**Table 4.11:** 0.9% NaCl based saline solution electrical properties

will be read by the electrode, and because it's the medium in which the signal sources are immersed. Being this material a biological material, its characterization is not trivial and it's been decided to use a frequency-dependant material characterization to be as realistic as possible, also because the insulating properties of this material are not negligible.

Nerve tissue is a non-isotropic material because of the topological organisation of the nerve structure (cfr. 3.2.1), being composed of nerve cells but also by other kinds of tissues like connective tissue, etc.

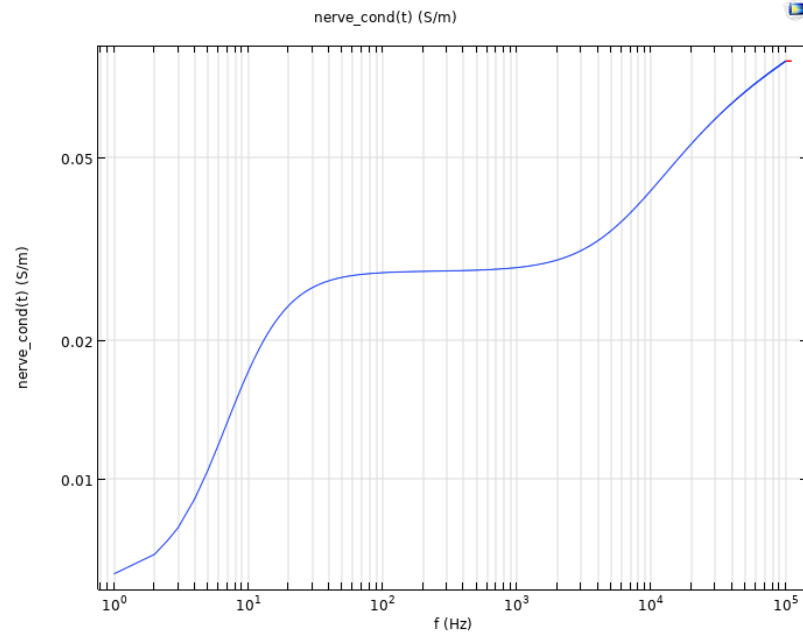
The presence of myelin and epineurium that is highly resistive causes a higher conductivity in the longitudinal direction. In this case, given the necessity of having a first attempt in short time, the characterization by IT'IS foundation was exploited [53]. These values were chosen because they are complete and are the same used in the commercial Sim4Life platform. Moreover, the nerve was modelled as a bulk material, neglecting the internal structures composing it and generating anisotropy. From the database, 1000 values between 0 and 1 kHz and 1000 between 1kHz and 100 kHz for both conductivity and permittivity were taken, and imported into COMSOL thanks to the interpolation function. For allowing faster studies, the values of the function at 1kHz, the order of magnitude of the peak of the power spectral density of the ENG signal, were extrapolated. They are reported in table 4.12.

Nerve tissue			
<i>Property</i>	<i>Variable</i>	<i>Value</i>	<i>Unit</i>
Electrical conductivity	$\sigma$	0.03 @1kHz	S/m
Relative permittivity	$\epsilon_r$	70000 @1kHz	-

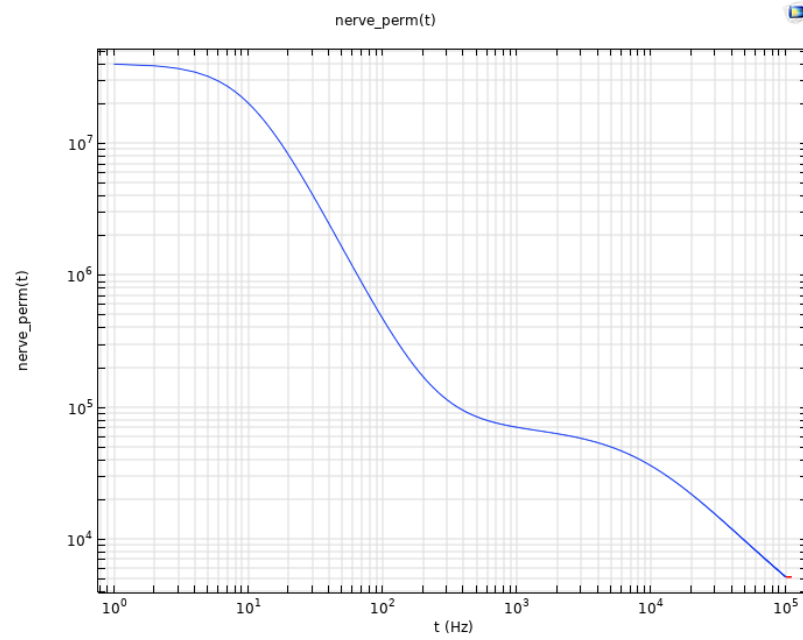
**Table 4.12:** Nerve tissue electrical properties at 1kHz

### 4.3.5 Physics

The objective is to study the propagation of the currents across the system and the generation of electrical potentials. More specifically, the need is to be able to assign to each NOR, modelled as a point, a current source that generates a



(a) Plot of nerve conductivity values



(b) Plot of nerve permittivity values

**Figure 4.7:** Nerve electrical properties from 0 to 100kHz, retrieved from [53]



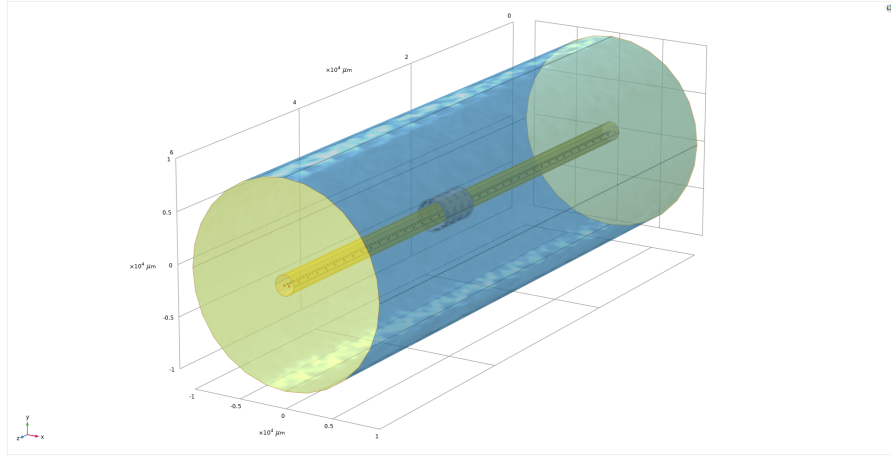
current defined by the outputs of the NEURON simulation, saved in appropriate files. That time dependant current should be appropriately imported and assigned to the corresponding NOR. This kind of study needs to be performed with the AC/DC module, and for the purpose, *Electric Currents* physics has been chosen [54].

### Conditions

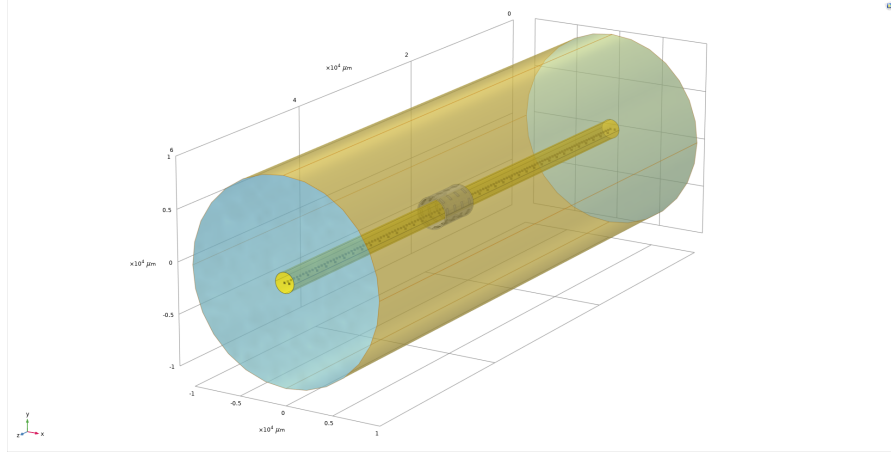
The physics *AC/DC - Electric currents* has been added to the whole model. When inserting this physics, the software automatically generates the following conditions: *Current Conservation* and *Initial Values*, applied to the whole volume; *Electric Insulation*, applied to the outer surface. The conditions have been customised as follows, to fit the simulation needs. First of all, *Electric Insulation* is applied to the bases of the cylinders representing the nerve and the saline solution, as in figure 4.8a. The simulation, in fact, concerns a tract of the nerve, on which we are interested in reading the signal transversally and "far enough" from the edges. Then, the current generated by the axons is assumed to disperse towards the outside. Therefore, the *Ground* is assigned to the outer surface of the cylinder, as in figure 4.8b. Finally, *Point Current Sources* are created and assigned to each point representing the NOR in the model. This process is not trivial and is described in detail below.

### Creation of current sources

Each NOR requires to generate the current that has been extracted from the NEURON simulation. Refreshing what has been described in section 4.2.1, NEURON simulations computes by default, for a specified section, a current density. In fact, the sections are modelled as cylinders, and the generation of current is regarded as a distributed mechanism on their surface rather than a point process. In COMSOL, however, NORs are modelled as points, so the currents need to be considered as point processes. The conversion was computed directly before generating the NEURON output files. First of all, the files containing currents must be imported into COMSOL. In COMSOL, it is possible to define custom functions in several ways, among which there are the *Analytic*, *Interpolation* and *Piecewise* function definitions. In this case, the most appropriate method revealed to be the use of *Interpolation*, by which it is possible to provide the points on which data are obtained, and decide an interpolation method to build a continuous function. With this method, currents for each NOR have been imported in *Global definitions* as Interpolation functions from files and grouped for each fibre type. Then, the currents have been inserted in the *Physics*, assigning them to point current sources located in the NOR points, by declaring the name of the function to be inserted into  $Q_{j,p}$ , the parameter that allows to set the current value. For the feasibility of this



(a)



(b)

**Figure 4.8:** Conditions imposed on the physics of the model, highlighted in yellow. In 4.8a, the bases set as electric insulation are shown, while 4.8b represents the surface of the saline medium declared as ground.

process, after the tests done importing the currents just in a few points, the need of automation implied the already mentioned use of the *Application builder*. After being inserted, the point current sources have been grouped together according to the axon they belong to.

#### 4.3.6 Study configurations and generation of the results

The objective of the simulation is to record electrical potential on the electrode contacts over time. Therefore, the study to be chosen is *Time Dependent*. When

setting a time-dependent study in COMSOL, it's necessary to decide the time range and the steps required for performing the simulation. The time range must be chosen according to the relevant window in which meaningful signals, namely the spikes, appear in each node of Ranvier. The time steps are identified by checking the  $dt$  in the NEURON simulation and selecting a comparable value, that allows not to lose relevant information, but possibly slightly higher, not to compromise computational efficiency. For all the simulations presented, the time interval is between 1 and 4 ms, with time steps of  $10\ \mu\text{s}$ . The computed results are saved in the dataset, and used mainly in two ways: first, create 3D plots and corresponding animations, second, export potential on the contacts as files, for further processing.

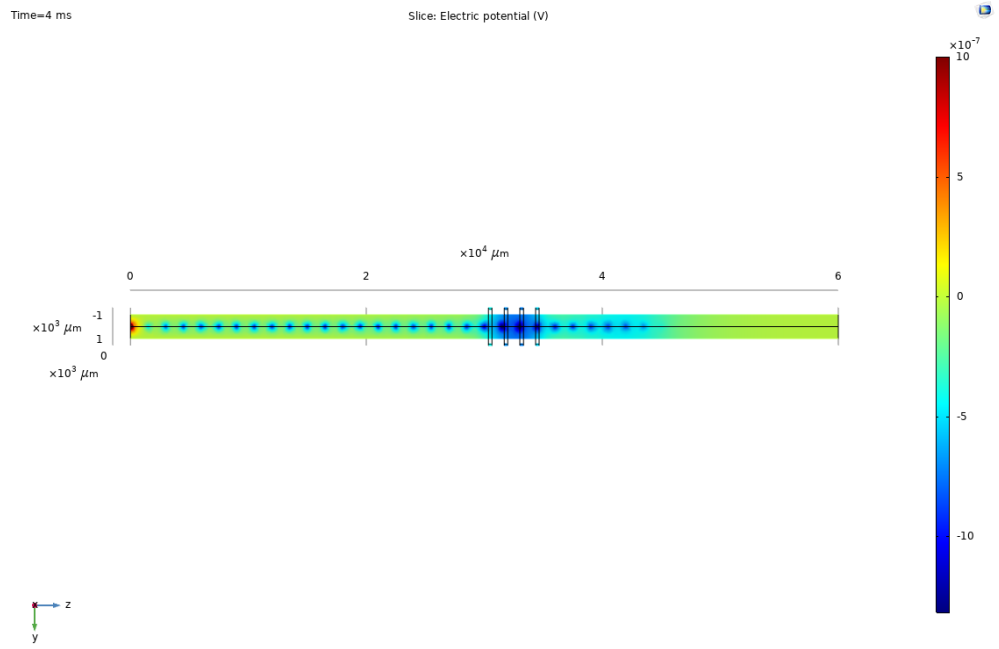
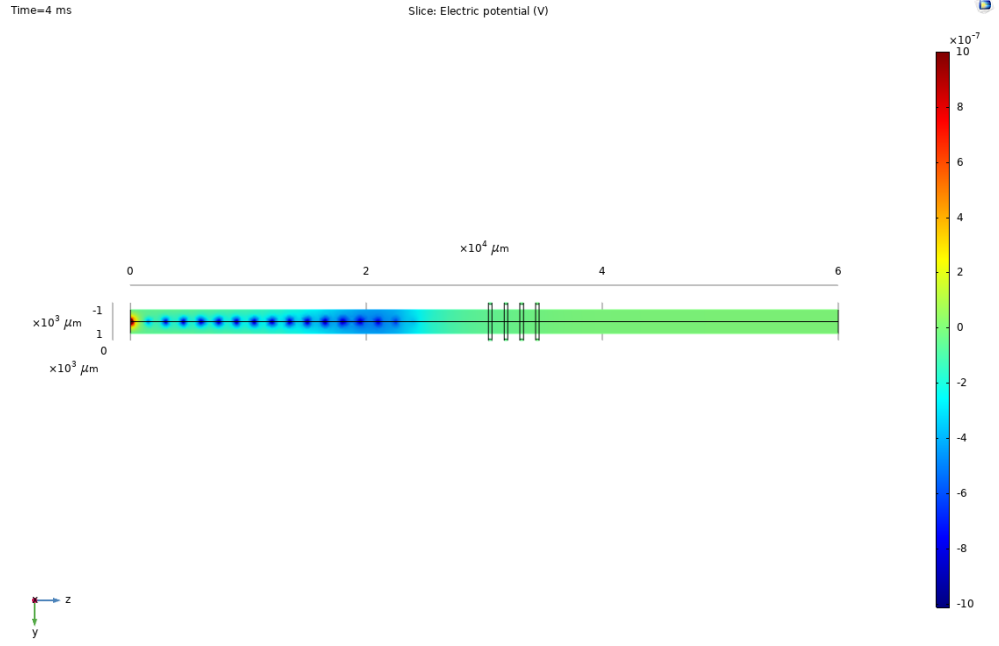
### 3D Plots generation

3D Plots are created mainly for visualising the propagation of the signals along the nerve at a selected time step, either representing its volume, surface or slices, and the signal sensing on the electrodes. For the models created in this work, *Electric potential* plots were chosen. On the nerve, volume or surface representations are useful for seeing what happens on what, in a more complex model, would be the epineurium, that is the external membrane of the nerve. Therefore, these plots show the potential at the interface between the nerve and the conductive medium, giving an interesting overview of the propagation of the APs. Slice plots are useful, instead, for visualising more closely the sequential activation of the NORs inside the nerve, allowing to verify the propagation of the potential across the single fibres and to highlight the variation of potential in the NOR neighbourhoods and its propagation across the volume, towards the outer surface. An example of slice plot is shown in figure 4.9. On the metal contacts, volume plots are used to visualise the generation of the recorded CAP, after the potential propagates from the nerve surface across the conductive medium, finally reaching the metal, and to see how every electrode senses the variation of potential with respect to the others, in terms of amplitude and delay, like in figure 4.10. Also, starting from these plots, animations can be produced, to show the sequence of frames as a video, providing an overview of the whole process.

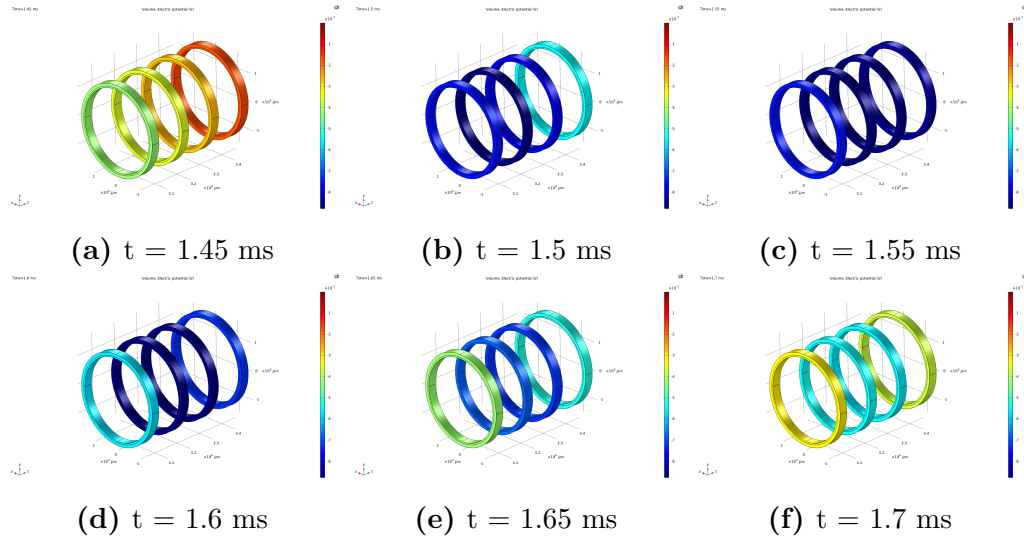
Examples of COMSOL 3D plots of a complete simulation are shown in figure 4.11.

### Electric potential data export

For pursuing the end objective of the simulations, it's necessary to read the voltage values on the electrode contacts and make them available for further processing. These values have been retrieved as volume averages on the single contact domain. After evaluation, a table is saved to *.txt* file and updated. This procedure is repeated for all the contacts, and, in the case of MCC, also an average on all the contacts over the same ring is computed, in order to provide a quick comparison

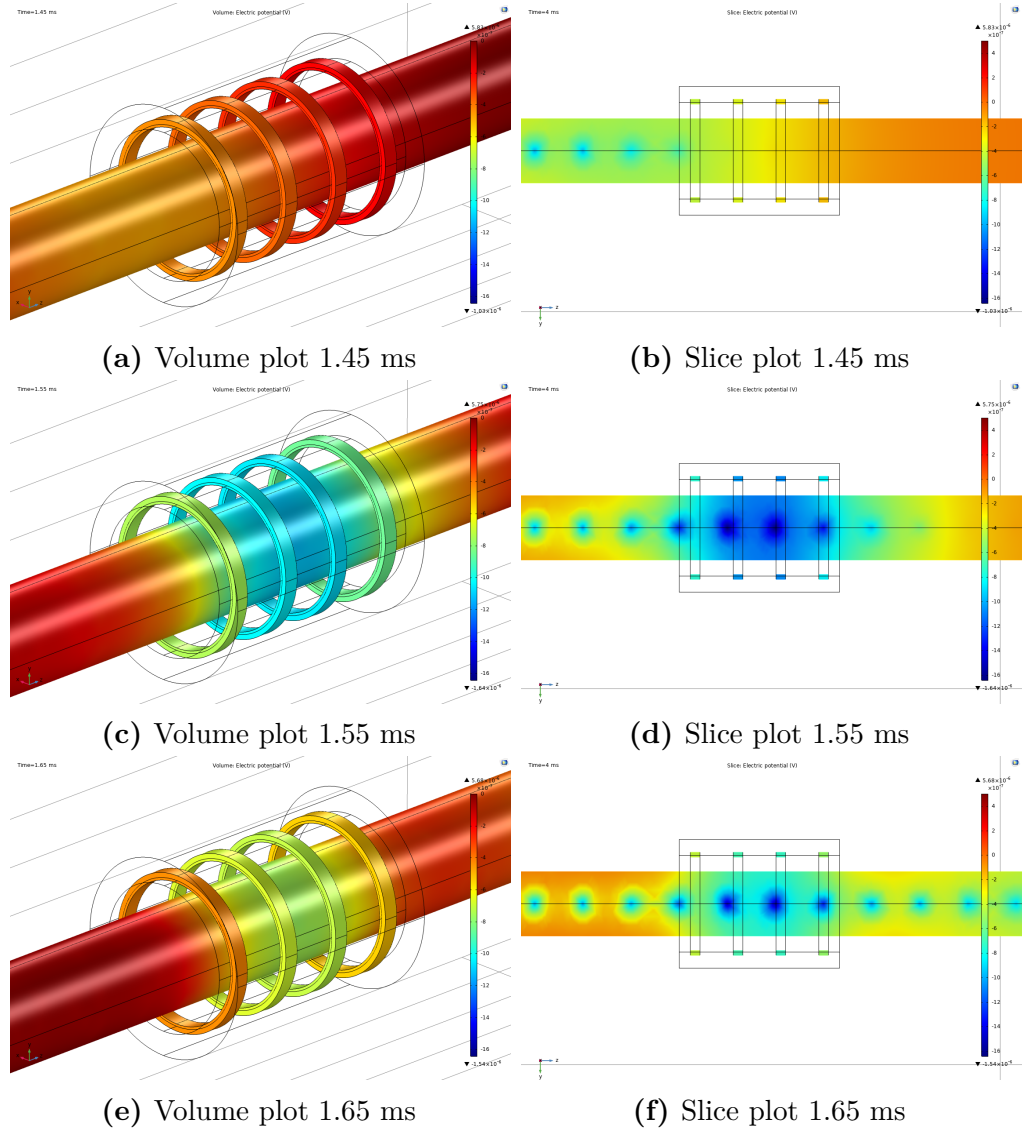


**Figure 4.9:** Example of slice plots, representing the propagation of an action potential in a 1-fibre nerve model.



**Figure 4.10:** Example of volume plots on the rings, showing the propagation of an action potential as read by the electrode.

between different configurations having the same number of rings. Also this process is automated by an appropriate function in the Application builder, to make it feasible on all the electrodes. Electrode contacts have been previously selected and renamed to easily go through all of them and create output files automatically.



**Figure 4.11:** Example of volume and slice plots of the nerve and electrode rings, showing the propagation of an action potential.

# Chapter 5

## Results

### 5.1 Output of the Neuron simulations

The simulations on NEURON were used to create the current data to be imported into the COMSOL simulation software. As soon as they were produced, to ensure that the output was the desired one, the currents were imported and processed into MATLAB (see code listed in Appendix A.3.1). The current outputs were characterized to evaluate their amplitude and to measure the conduction velocity along the axon. The results of these measurements are presented here in the form of graphs and tables.

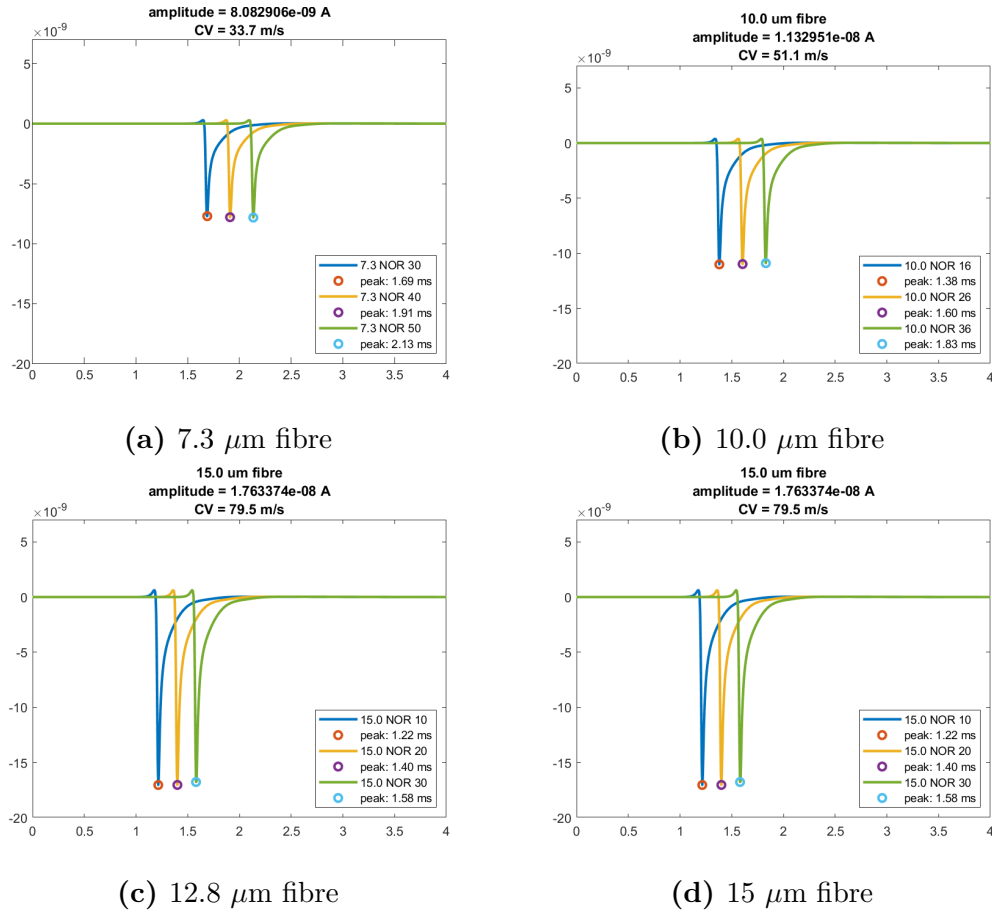
The graphs shown in figure 5.1 show for some exemplified simulated fibres, the waveforms obtained in the central NOR of the fiber, at the tenth nodes preceding and following it. The peak values (in module, as the waveform has a negative main peak) have been identified and indicated on the graph, and are those used for the calculation of the conduction velocity.

In the table 5.1, the CVs obtained for each fiber are presented, together with the amplitude value of the measured current.

#### Remarks on NEURON simulations

What is observed from the results of the simulations is that the amplitude values of the membrane currents increase with an increasing fiber diameter. This outcome is in agreement with the experimental observation: in the recording of ENG signals, the fibers with a larger diameter give a more significant contribution to the recorded signal. Starting from this observation, it is expected to obtain, in the COMSOL simulations, higher values of recorded potential in the simulations containing large caliber fibres.

The peak-to-peak value of the current generated in a NOR in the smallest simulated fiber, that is  $5.7 \mu\text{m}$  in diameter, is one third of the amplitude recorded in the



**Figure 5.1:** Examples of the plot of the currents in the central Node of Ranvier, the the previous tenth NOR and the following tenth NOR, simulated in the NEURON software and used for computing the CV and the current amplitude.

largest one, that is  $16\mu\text{m}$ .

The signals have comparable morphology. The central signal, represented in yellow, is that of the mid-axon node. The central nodes are located approximately at the same spatial position for all axons, around 3cm from the stimulation node. Although the graphs do not take spatial information into account, this consideration allows you to appreciate the effect of the different conduction speeds on the graphs. In fact, it is qualitatively noted how the signal arrives at about half of the axon in advance as the diameter of the fiber increases.

The CVs, estimated and reported in the table, confirm this qualitative observation. The values obtained are in accordance with the experimental measurements of the conduction velocities of the myelinated fibers, presented in the section 3 in table 2.1.



CV and amplitude for simulated fibres		
<i>Diameter (<math>\mu m</math>)</i>	<i>CV (m/s)</i>	<i>Amplitude (nA)</i>
5.7	23.5	6.22
7.3	33.7	8.08
8.7	44.0	9.83
10.0	51.1	13.24
12.8	65.9	14.65
14	72.7	16.26
15	79.5	17.63
16	85.7	18.75

**Table 5.1:** Computed CVs and current amplitudes for the 9 types of fibres simulated within NEURON.

Here too, the conduction velocity increases with increasing fiber diameter, presenting a minimum value of 23.5 m/s in the smallest simulated fiber and a value of 85.7 m/s in the largest one.

The compliance of the values obtained with the experimental results makes the axons simulated with NEURON an excellent building block for the construction of the nerve model, confirming what has been found in the literature.

## 5.2 Preliminary simulations

The preliminary simulations are aimed at evaluating the influence of the basic parameters of the model on the resulting recorded waveform. They are based on very simple models, containing one motor fiber within the cylindrical structure of nervous tissue. They are aimed at verifying, in particular, the influence of the geometric parameters and the definition of materials on the results and on computational time, in order to provide a reference point for future optimizations, to arrive at more complex models.

The aspects tested include the influence of the nerve diameter, the thickness of the cuff, the definition of the nerve material, the dimension of the ground cylinder, and the effect of different meshes. The starting parameters, which undergo changes during the analysis, are summarized in the table 5.2.

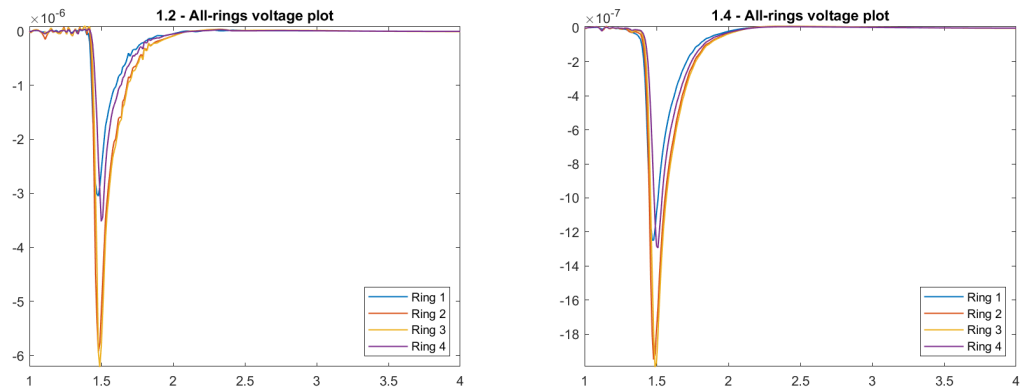
### 5.2.1 Effect of fibre diameter

For the first simulations, the diameter of the fiber was tested, to see the attenuation effects of the thickness of the material interposed between sources and electrodes. The comparison was conducted on 16  $\mu m$  fibers, placed in the center of the nerve,

Overall parameters - preliminary simulations		
	Nerve diameter	
	1 mm	2 mm
<b>Cuff model</b>		
Length	5 mm	5 mm
Diameter	1.5 mm	3 mm
Thickness	250 $\mu\text{m}$	500 $\mu\text{m}$
IED	1.333 mm	1.333 mm
Edge dim.	500 $\mu\text{m}$	500 $\mu\text{m}$
<b>Complete model</b>		
Fibre coordinates	(0,0)	(0,0)
Nerve tissue model	Simple isotropic	Simple isotropic
Mesh	Finer	Finer
GND cylinder diam.	10 mm	20 mm

**Table 5.2:** Starting parameters for the preliminary simulations

representing alpha motor neurons which are among the main components of the ENG signal due to their greater amplitude. The results of the reading on the electrodes are shown in figure 5.2, the peak values of the signals are shown in table 5.3.



**Figure 5.2:** Potentials recorded from nerves with different diameters.

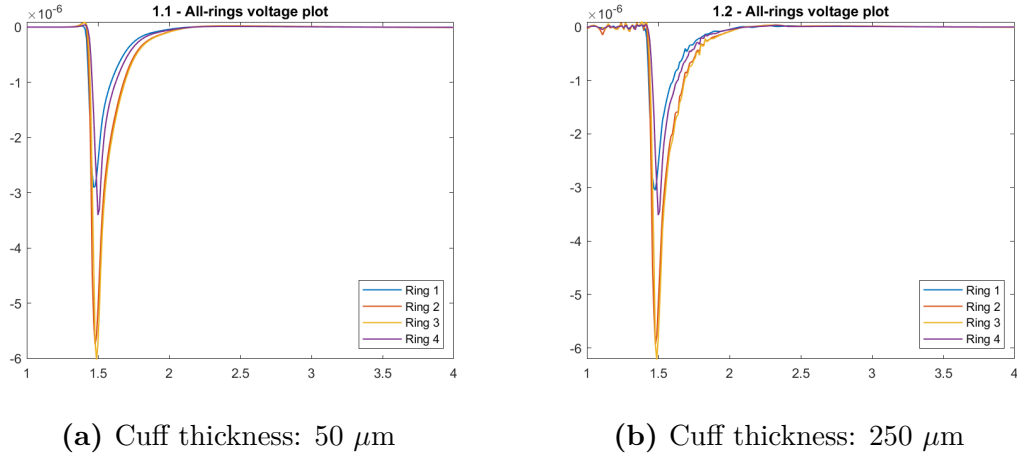
It can be seen that values almost triple passing from 2mm to 1mm nerve diameter.

Effect of nerve model diameter on recorded potential (V)		
	1 mm	2 mm
Ring 1	3.1e-06	1.2562e-06
Ring 2	6.0273e-06	1.9593e-06
Ring 3	6.3049e-06	2.0034e-06
Ring 4	3.5624e-06	1.3008e-06
Average	4.7487e-06	1.6299e-06

**Table 5.3:** Peak-to-peak amplitudes on the electrode rings and their average in the recording from nerves with different diameters.

### 5.2.2 Effect of the thickness of the cuff

The second test was carried out on the same models of the previous tests using 5 times thinner cuffs, to test the effect on the recording of the thickness of the insulation layer. The 1 mm nerve fibre has been tested with cuffs of 50  $\mu\text{m}$  and 250  $\mu\text{m}$  thickness. The 2 mm nerve fibre, has been tested with cuffs of 100  $\mu\text{m}$  and 500  $\mu\text{m}$  thickness. The results of the simulations are shown in figure 5.3 for the 1 mm nerve fibre and in figure 5.4 for the 2 mm nerve fibre. The values of the peak of the potentials read from the cuff are presented in table 5.4 for the 1 mm fibre and 5.5 for the 2 mm fibre.

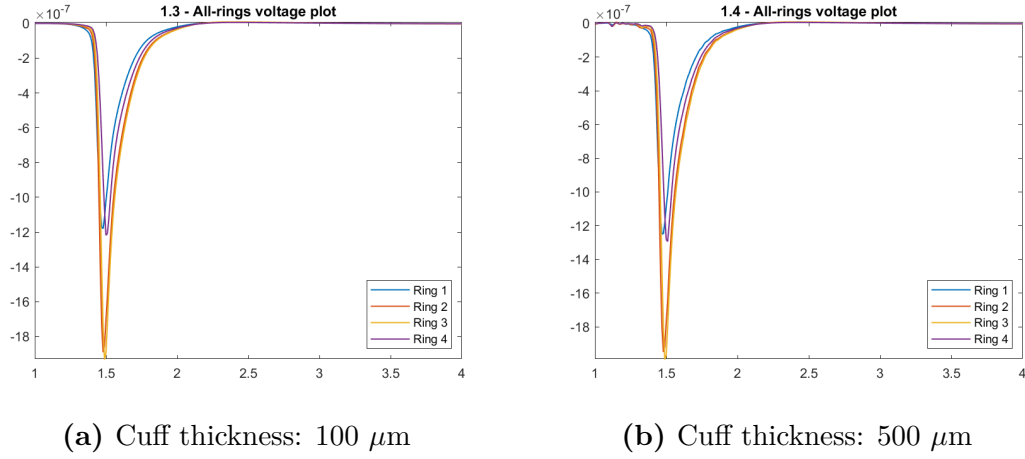


**Figure 5.3:** Potential read on the 4 rings of a MEC on a 1 mm fibre with 1.5 mm diameter cuffs with different thickness.

The expected result was that a thicker cuff, having a larger insulating layer, would provide higher voltage values. Counterintuitively, both from the figures and the

Effect of the cuff thickness on recorded potential (V)		
1 mm nerve model		
	50 $\mu m$	250 $\mu m$
Ring 1	2.9206e-06	3.1e-06
Ring 2	5.8386e-06	6.0273e-06
Ring 3	6.1138e-06	6.3049e-06
Ring 4	3.4468e-06	3.5624e-06
Average	4.58e-06	4.7487e-06

**Table 5.4:** Peak-to-peak amplitude on a 1 mm fibre with 1.5 mm diameter cuffs with different thickness.



**Figure 5.4:** Potential read on the 4 rings of a MEC on a 2 mm fibre with 3 mm diameter cuffs with different thickness

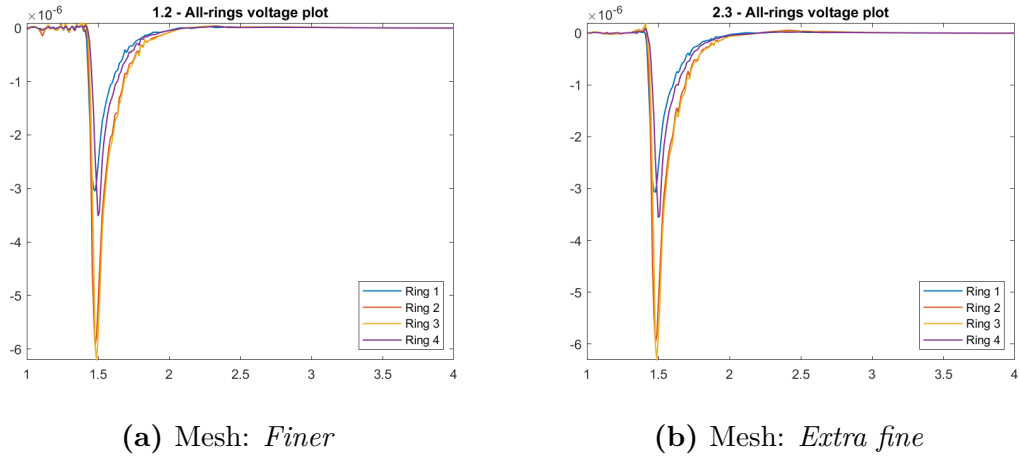
tables, it is observed that the recorded values do not change appreciably, neither in the 1 mm nor in the 2 mm nerve fibres. The peaks of the signals have similar values. From the observations, it therefore seems that even the thinnest cuffs already have a high enough level of electrical insulation to force currents to flow towards the ground through the only fluid interposed between the nerve and the electrodes, providing, as a consequence, similar values to their thicker version. However, it must be noticed that the recordings with a thinner cuff electrode, all other parameters being equal, provide a less noisy signal. To this end, it can be said that thinner cuff electrodes provide better results and should be preferred in the future simulations.

Effect of the cuff thickness on recorded potential (V)		
2 mm nerve model		
	100 $\mu m$	500 $\mu m$
Ring 1	1.1861e-06	1.2562e-06
Ring 2	1.8998e-06	1.9593e-06
Ring 3	1.9417e-06	2.0034e-06
Ring 4	1.2242e-06	1.3008e-06
Average	1.563e-06	1.6299e-06

**Table 5.5:** Peak-to-peak amplitude on a 2 mm fibre with 3 mm diameter cuffs with different thickness.

### 5.2.3 Effect of the mesh

The second test is carried out with different types of mesh, to see if using a finer one and consequently increasing computational times, will benefit in terms of simulation quality. On both 1 mm and 2 mm nerves, mesh options *Finer* and *Extra fine* have been tested. The comparison of the two types of mesh is shown in figure 5.5 for the 1 mm nerve diameter and in figure 5.6 for the 2 mm nerve diameter. The peak values of the electric potential on the rings are presented in table 5.6 for the 1 mm nerve and in 5.7 for the 2 mm nerve. The recorded values are once again

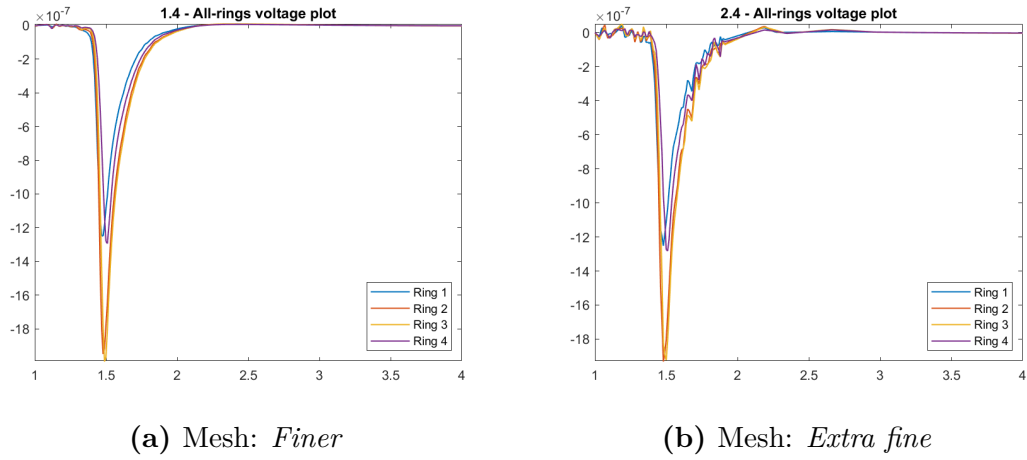


**Figure 5.5:** Potential on a 1 mm fibre with different mesh settings

very similar to each other. Comparing the graphs, no improvement is observed on signal quality: on the contrary, the signal appears more disturbed than the corresponding versions with finer mesh, for computational reasons, attributable

Effect of the mesh on recorded potential (V) 1 mm nerve model		
	<i>finer</i>	<i>extra fine</i>
Ring 1	3.1e-06	3.1044e-06
Ring 2	6.0273e-06	6.1511e-06
Ring 3	6.3049e-06	6.5114e-06
Ring 4	3.5624e-06	3.6502e-06
Average	4.7487e-06	4.8543e-06

**Table 5.6:** Peak-to-peak amplitude on a 1 mm fibre with different mesh settings



**Figure 5.6:** Potentials on a 2 mm fibre with different mesh settings

to the convergence of solutions. The computational times, going from *Finer* to *Extra fine* mesh, range from about 6 minutes to about 13 minutes. Since there is no improvement in performance and increasing the time required for simulation, the choice of a *Finer* mesh is to be preferred.

#### 5.2.4 Effect of the nervous tissue model

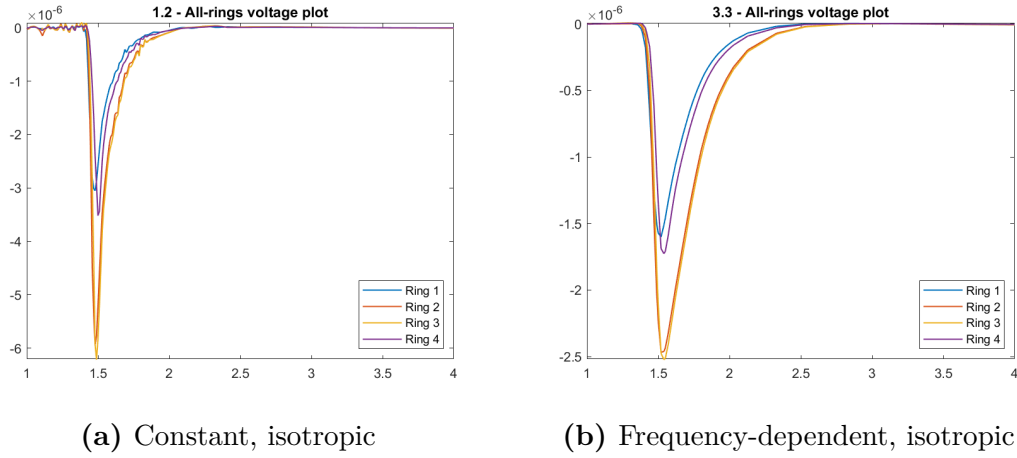
For the purpose of these simulations, the components of the nervous tissue have been modeled as isotropic, for simplicity of definition, remembering however that the nerve, as stated in chapter 3, is anisotropic, in the sense that it presents a transverse resistivity that is greater than the longitudinal one.

In the hypothesis of nerve isotropy for the simulations conducted, the values for relative permittivity and conductance were retrieved from the IT'IS Foundation website, which provides a database for the modeling of biological tissues and which

Effect of the mesh on recorded potential (V) 2 mm nerve model		
	<i>finer</i>	<i>extra fine</i>
Ring 1	1.2562e-06	1.2941e-06
Ring 2	1.9593e-06	1.9771e-06
Ring 3	2.0034e-06	1.9754e-06
Ring 4	1.3008e-06	1.3061e-06
Average	1.6299e-06	1.6382e-06

**Table 5.7:** Peak-to-peak amplitude on a 2 mm fibre with different mesh settings

is the basis of the simulation platform Sim4Life [55], as shown in chapter 4. The objective of the simulations presented below is to quantify the effect of a frequency-dependent modeling of the tissue with respect to a modeling with constant values of relative permittivity  $\epsilon_r$  and conductivity  $\sigma$ . Once again, the simulations were carried out on nerves with diameters of 1 mm and 2 mm. For the first diameter, the results are presented in table 5.8 and in the graphs in figure 5.7. For the second, refer to table 5.9 and figure 5.8.



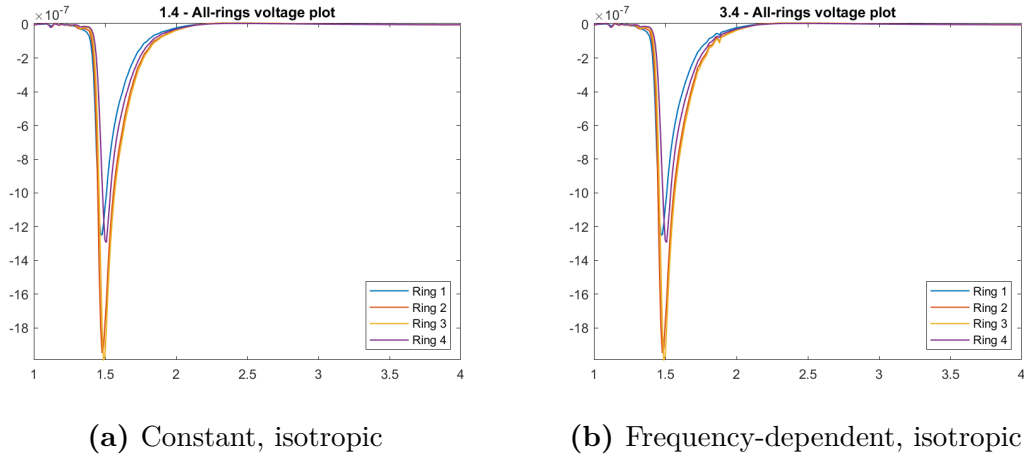
**Figure 5.7:** Signals on a 1 mm nerve with different definitions of the nerve tissue electrical properties

### 5.2.5 Effect of ground dimension

The effect of the conductive medium dimension on the potentials is the last aspect tested in these preliminary simulations. In particular, the test was conducted on

Effect of nerve tissue electrical properties on recorded potential (V)		
1 mm nerve model		
	<i>Constant</i>	<i>Freq.-dependent</i>
Ring 1	3.1e-06	1.6056e-06
Ring 2	6.0273e-06	2.4708e-06
Ring 3	6.3049e-06	2.531e-06
Ring 4	3.5624e-06	1.7312e-06
Average	4.7487e-06	2.0846e-06

**Table 5.8:** Peak-to-peak amplitude on a 1 mm nerve with different definitions of the nerve tissue electrical properties



**Figure 5.8:** Peak-to-peak amplitude on a 1 mm nerve with different definitions of the nerve tissue electrical properties

the smaller diameter (1mm) nerve model.

In the simulations presented up to now, the diameter of the cylinder representing the saline solution, that is the one on which the ground condition is imposed, has always been fixed as 10 times the diameter of the studied nerve. This implies that the dimensions of this cylinder, for the simulation of 2mm and 1mm nerves are obviously different.

The objective of the test presented here is to evaluate how the potentials sensed change if, for a 1 mm fiber, the cylinder diameter is brought to the value used for 2 mm fibers.

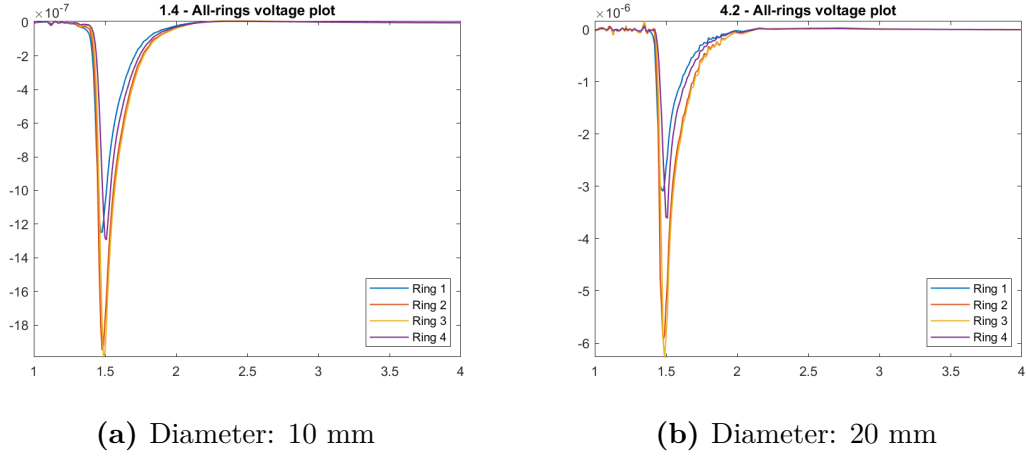
Results of the simulation are presented in table 5.10 and in figure 5.9.

In the second case, a slight increase in values would be expected. This derives



Effect of nerve tissue electrical properties on recorded potential (V) 2 mm nerve model		
	<i>Constant</i>	<i>Freq.-dependent</i>
Ring 1	1.2562e-06	1.256e-06
Ring 2	1.9593e-06	1.959e-06
Ring 3	2.0034e-06	2.003e-06
Ring 4	1.3008e-06	1.3005e-06
Average	1.6299e-06	1.6296e-06

**Table 5.9:** Peak-to-peak amplitude on a 1 mm nerve with different definitions of the nerve tissue electrical properties



**Figure 5.9:** Signals on a 2 mm nerve with different diameters of the cylinder set as ground.

from the observation that, with a fixed current, maintaining constant the resistance (correlated to the thickness) of the nervous tissue and varying that of the saline solution, there is a greater voltage drop on the saline solution which leads to an increase in the recorded voltage.

Obviously, the problem is much more complex and these are only rough considerations. Indeed, this occurs to such a small extent (1.85% of the average value on the 4 electrodes), that it must be considered negligible.

Effect of the diameter of the GND cylinder on recorded potential (V)		
1 mm nerve model		
	10 mm	20 mm
Ring 1	3.1e-06	3.1618e-06
Ring 2	6.0273e-06	6.0646e-06
Ring 3	6.3049e-06	6.4389e-06
Ring 4	3.5624e-06	3.6811e-06
Average	4.7487e-06	4.8366e-06

**Table 5.10:** Peak-to-peak amplitude on a 1 mm nerve with different diameters of the cylinder set as ground.

## 5.3 Single axon models

### 5.3.1 Effect of the cuff geometry

The experiments on single fiber were carried out to give a first example of use of the created tool. In particular, they were made to test the influence of the geometric parameters of the cuff on the quality of the signals recorded. Simulations in this section start with a test on the *Edge\_dist* parameter of the electrodes, and continue with tests done varying the cuff diameter, in order to assess the effect of the distance of the outermost electrodes from the ends of the cuff and of the amount of physiological solution interposed between the electrode surface and the nerves.

#### Effect of the dimension of the cuff edges

As visible from the preliminary simulations, the outermost electrodes show a smaller amplitude of the recorded potential. This phenomenon can be explained by the greater proximity of the contact to the end of the cuff structure. Remembering that the cuff has an insulating effect and that it is used to confine the nerve potentials and allow their recording at the highest possible amplitudes, the outermost electrodes, being close to the cuff openings, are less affected by this effect, the potential is more dispersed and consequently the sensed voltage is lower.

The simulations were conducted on 16  $\mu\text{m}$  fiber in 2 mm nerve, with a 3 mm diameter cuff. The inter-electrode distance was kept constant, equal to 1.333 mm, by varying the length of the cuff and the distance from the edges as indicated in table 5.11, that summarizes the results obtained, together with figure 5.10.

To better quantify the variation, the mean of the peak-to-peak amplitudes was calculated, together with the variance of the peak-to-peak amplitudes normalized

with respect to the mean.

<b>Effect of the dimension of the cuff edges on recorded potential (V)</b>			
	<i>5 mm cuff</i>	<i>6 mm cuff</i>	<i>7 mm cuff</i>
	<i>0.5 mm edges</i>	<i>1 mm edges</i>	<i>1.5 mm edges</i>
Ring 1	1.19e-06	1.71e-06	2.34e-06
Ring 2	1.90e-06	2.36e-06	3.01e-06
Ring 3	1.94e-06	2.42e-06	3.01e-06
Ring 4	1.22e-06	1.77e-06	2.48e-06
Average	1.56e-06	2.06e-06	2.71e-06
Std	0.2648	0.1833	0.1305

**Table 5.11:** Signals recorded by varying the the dimensions of the outer edges of the cuff (distance between the outer electrodes and the ends of the tube) keeping IED constant.

It is observed that the increase in distance is accompanied by more uniform values of the recorded potentials. The increase in the length of the cuff, constituting a greater obstacle to the flow of electric current, causes a significant increase in the average peak value of the signal.

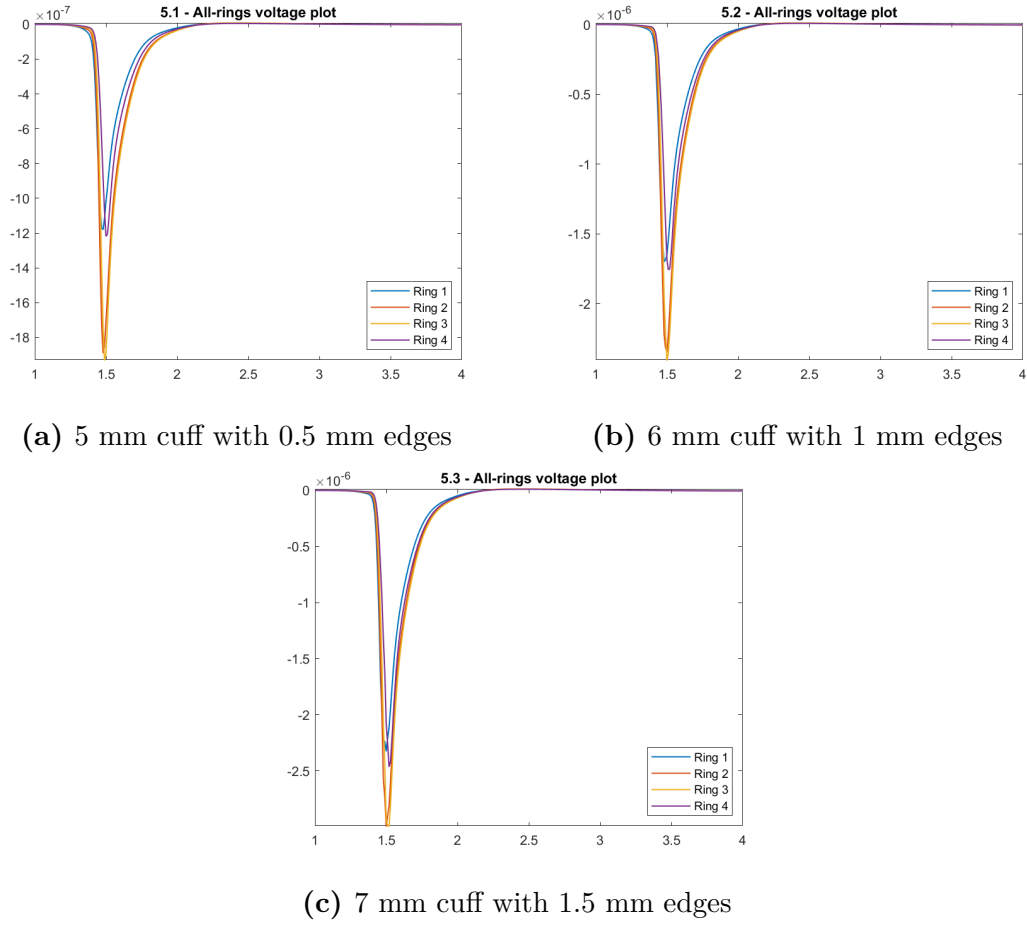
### Effect of the fluid between the nerve and the cuff

The effect of the amount of liquid between nerve and cuff was tested to give an idea of how this parameter can affect the simulation results and, even more, how this can have an influence on the quality of the results obtained in the experimental case.

The amount of liquid is responsible for the poor quality of the recorded signal. In the ideal case, we would like to have a perfect contact between electrode and nerve. This is not possible as the cuff, for biocompatibility reasons, must necessarily be of a larger diameter than that of the nerve, as already explained in the background section, leading to a short-circuit effect that reduces the performance of the system, making the voltages sensed by the different electrodes very similar to each other. Furthermore, in the experimental case it is unthinkable to have the two cylinders, nerve and cuff, perfectly coaxial. This brings to impedance imbalances, which cause a further deterioration in the performance of the recording system.

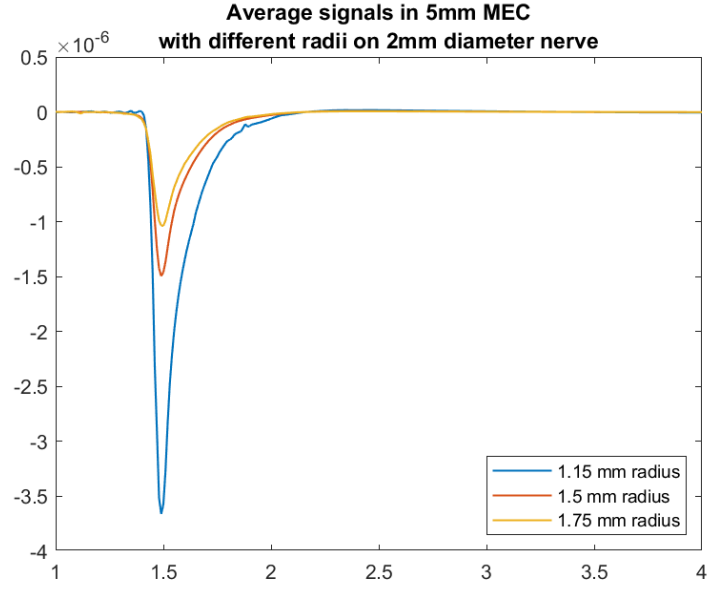
From the simulations it is expected that an increase in the diameter of the cuff, here exaggerated, leads to a greater overlap of the signals and a lower amplitude, with consequent worsening of the recording quality.

The results are shown in figure 5.11. In table 5.12, the average amplitudes of the

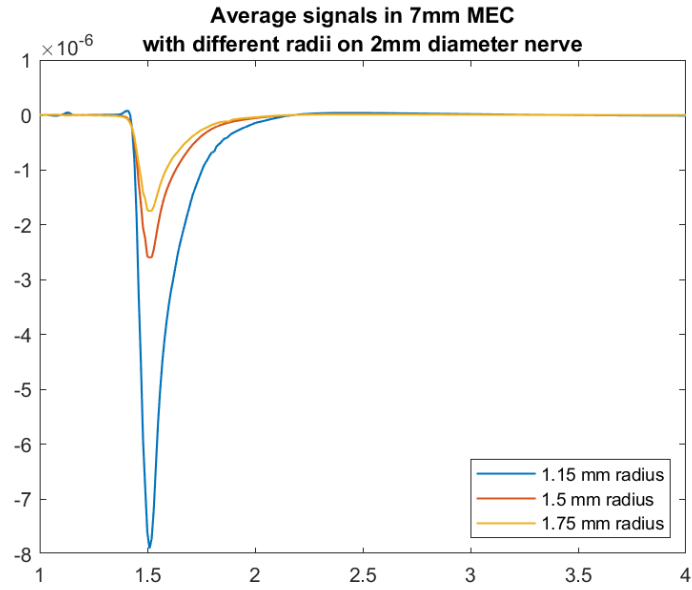


**Figure 5.10:** Signals recorded by varying the the dimensions of the outer edges of the cuff (distance between the outer electrodes and the ends of the tube) keeping IED constant

signals and their width, quantified with the Full Width Half Maximum (FWHM) calculated on the signal module, are summarized. To evaluate the quality of the signal, other than its amplitude, an aspect ratio has been calculated as the ratio between the FWHM and the signal amplitude. The higher the aspect ratio, the less the signal is sharp and therefore the poorer is its quality. From the data obtained, it is observed that the increase in the diameter of the cuff and consequently the greater quantity of interposed liquid worsen the quality of the signals, giving rise to less sharp action potentials, as visible from the considerable increase in the ratio between FWHM and amplitude. This is also accompanied by a markedly reduced amplitude, which in conditions of external noise could easily preclude the correct registration of the potentials. Results from the 7 mm cuff show higher values of



(a) Average signals on 5 mm MECs



(b) Average signals on 7 mm MECs

**Figure 5.11:** Effect of the edge dimension on the shape and the relative amplitude of the average signals sensed on the rings

Effect of the layer of conductive medium between nerve and cuff (cuff radius) on recorded potential (V)			
	1.15 mm radius	1.5 mm radius	1.75 mm radius
<b>5mm cuff</b>			
Average amplitude	3.83e-06	1.56e-06	1.08e-06
FWHM (ms)	0.10	0.11	0.13
FWHM/ $A_{pp}$	2.61e+04	7.05e+04	1.2037e+05
<b>7mm</b>			
Average amplitude	8.16e-06	2.71e-06	1.82e-06
FWHM (ms)	0.11	0.12	0.14
FWHM/ $A_{pp}$	1.34e+04	4.42e+04	7.69e+04

**Table 5.12:** Effect of the edge dimension on the waveform parameters of the average signals sensed on the different rings

potential. This is due to the stronger insulating effect of the longer cuff, that helps keeping the currents better confined within the structure.

### 5.3.2 Effect of the fibre position

In this section, the topological properties of the nerve start to be tested, i.e. those concerning geometrical distribution of the fibres within the section of the nerve structure. The first tests are carried out on single fibre models, evaluating the signals recorded with MEC and MCC electrodes. This will work as a base for understanding what happens in the case of multiple fibres dispersed across the nerve section, as it will be examinaed in the slightly more complicated models presented in section 5.4.

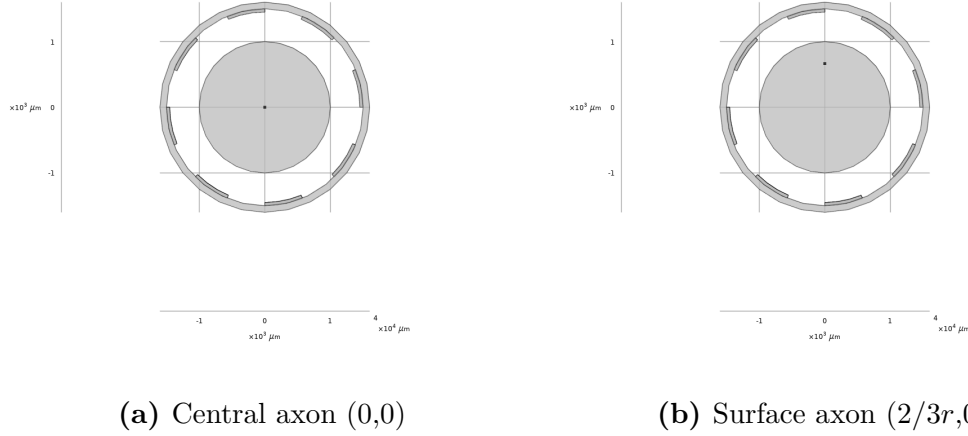
#### Effect of the position of a single fibre

In this series of tests, the effect on the signal of the displacement of a single fiber was evaluated, in particular when the fiber moves towards the external surface of the nerve. Tests were conducted with  $5.7 \mu\text{m}$  and  $16 \mu\text{m}$  fibres in 2 mm diameter nerves.

The position of the fibre has been varied in the coordinates  $(0,0)$ ,  $(r/3,0)$  and  $(2/3 * r,0)$ . The recordings were made first with MEC electrodes (length 5 mm, thickness  $1100 \mu\text{m}$ , inter-electrode distance 1.333 mm) and then with MCC electrodes with the same characteristics and 8 contacts per ring.

The first comparison presented is on a  $5.7 \mu\text{m}$  fiber located in the center of

the nerve, and the same fiber in a superficial position, in coordinates  $(2/3 * r, 0)$  in the nerve section. The two different configurations are shown in figure 5.12.



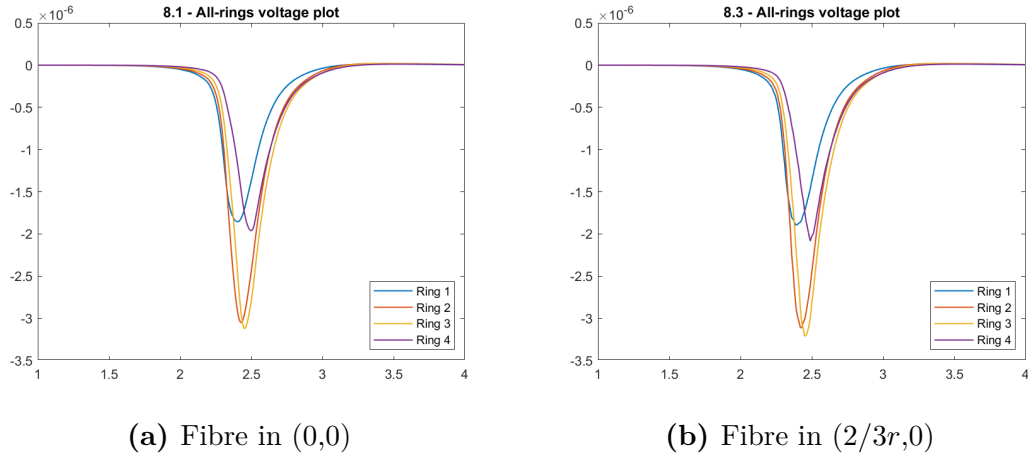
**Figure 5.12:** Section of the nerve showing different positions of the axon

The results of this simulation are compared in terms of amplitude, FWHM and waveform on the 3 rings, the first two summarized in table 5.13, the latter in figure 5.13. A perfect overlap of parameters and waveforms is observed. This confirms the

**Effect of the nerve fibre position  
on recorded potential (V)**

5.7 $\mu\text{m}$ fibre		
	<i>Central (0,0)</i>	<i>Surface (<math>2/3r, 0</math>)</i>
Ring 1	1.87e-06	1.47e-06
Ring 2	3.07e-06	2.20e-06
Ring 3	3.14e-06	2.29e-06
Ring 4	1.97e-06	1.63e-06
Average	2.51e-06	1.90e-06
FWHM (ms)	0.263	0.360

**Table 5.13:** Effect of the position of the nerve fibre across the nerve section on the peak-to-peak potential amplitudes and on the waveform FWHM, in the case of axons located centrally and towards the surface. Signals recorded with MECs.



**Figure 5.13:** Effect of the position of the nerve fibre across the nerve section on the potential amplitudes recorded with MECs in the case of axons located centrally and towards the surface.

thesis that, by means of a MEC, the spatial information on the arrangement of the sources within the nerve section is lost, keeping only the longitudinal information.

The same test was then performed using MCCs. Therefore, the recordings made with MEC and MCC on a  $5.7 \mu\text{m}$  fiber positioned first centrally in the nerve, and then at the coordinates  $(2/3r, 0)$  are compared below. For the comparison, first of all, the average waveforms on the 4 electrode rings are represented. In the case of MCC, the average waveforms on the 4 contact rings per contact are plotted.

A second plot represents the waveforms on each ring: in the case of the MCC these were obtained by averaging the values of all the contacts. The tables compare the results of the amplitudes found in the latter case.

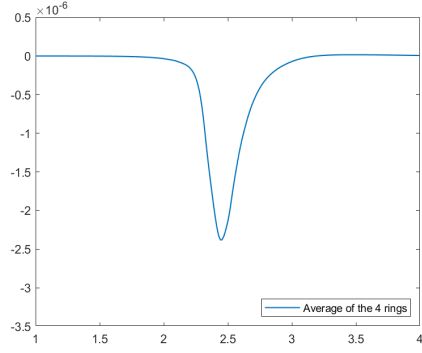
**MEC and MCC recordings on a centrally-located small fibre** For the case of a  $5.7 \mu\text{m}$  fiber located in  $(0,0)$ , the potentials averaged from the 4 rings recordings are shown in figure 5.14.

It is observed that, in the case of MCC, all the rings sense the same potential and the waveforms are therefore perfectly superimposed, in an equivalent way to what happens in the case of the MEC, but showing lower potential values.

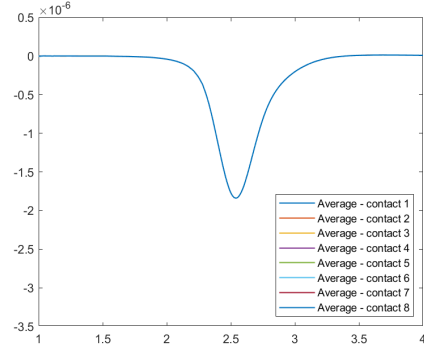
The voltage on each of the 4 rings, found by averaging the contacts in the case of the MEC, are shown in figure 5.15 and presented in table 5.14.

The value of potential recorded on a MCC and obtained by averaging the potentials on the single contacts, in this case, appears to give lower values than the potential read with a MEC. As visible in table 5.14 and in figure 5.15, the average voltage



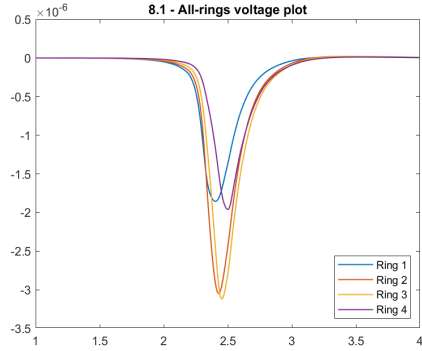


(a) Electrode type: MCC

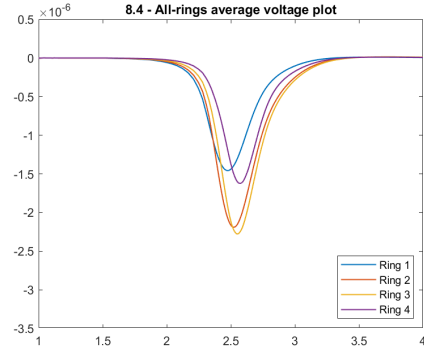


(b) Electrode type: MCC

**Figure 5.14:** Potentials from a  $5.7 \mu\text{m}$  centrally-located fibre averaged from the 4 rings, obtained different electrode types.



(a) Electrode type: MCC



(b) Electrode type: MCC

**Figure 5.15:** Potential values on each ring from a  $5.7 \mu\text{m}$  centrally-located fibre, obtained different electrode types.

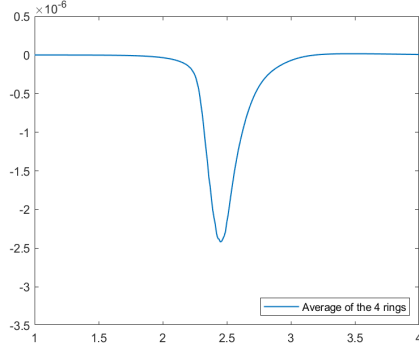
obtained with a MEC is about 33% higher than the one obtained with a MCC, while FWHM on MCC is 37% larger than in a MEC recording. This means that passing from a MEC to a MCC without exploiting the possibility of having a topological recording is not in the ideal case.

**MEC and MCC recordings on a surface-located small fibre** For the case of the same  $5.7 \mu\text{m}$  fiber moved into position  $(2/3 * r, 0)$ , the averaged potentials on the 4 rings are shown in figure 5.16.

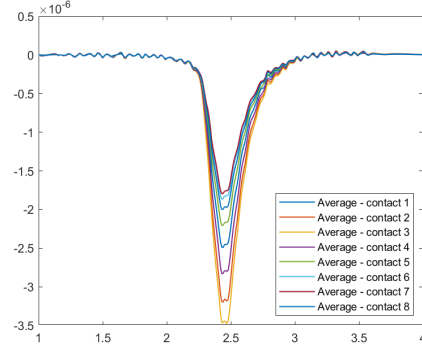
It is observed that, in the case of MCC, the rings that are close to the fibre record higher values of potential. This means that the MCC as expected, can provide

Effect of the electrode type on recorded potential (V) 5.7 $\mu\text{m}$ fibre in (0,0)		
	<i>MEC</i>	<i>MCC</i>
Ring 1	1.8687e-06	1.4676e-06
Ring 2	3.0719e-06	2.2048e-06
Ring 3	3.1423e-06	2.2928e-06
Ring 4	1.9748e-06	1.6337e-06
Average	2.5144e-06	1.8997e-06
FWHM	0.2630 ms	0.3600

**Table 5.14:** Peak-to-peak values, average and FWHM for the potential waveform from a 5.7  $\mu\text{m}$  centrally-located fibre, obtained different electrode types.



(a) Electrode type: MEC

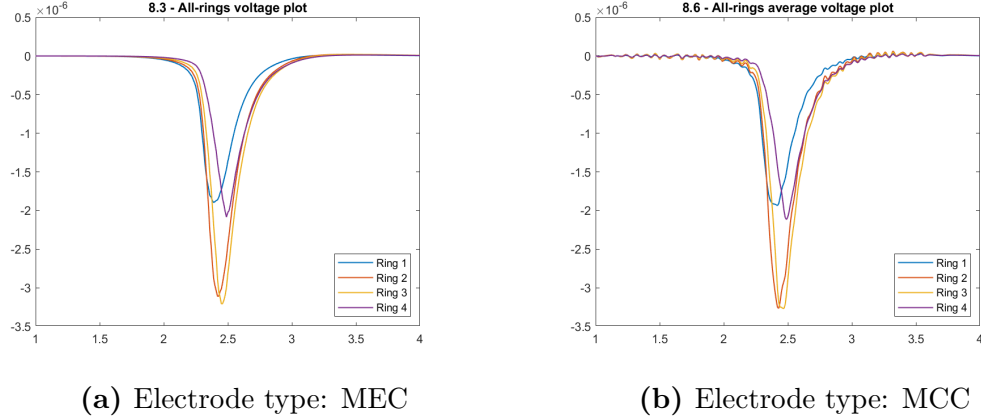


(b) Electrode type: MCC

**Figure 5.16:** Potentials from a 5.7  $\mu\text{m}$  surface-located fibre averaged from the 4 rings, obtained different electrode types.

information on the topological distribution of the fibres inside the nerve. The voltages on each of the 4 rings, found by averaging the contacts in the case of the MEC, are shown in figure 5.17 and presented in table 5.15.

Except for the noise component in the MCC recording, probably due to convergence aspects of the solver, the recorded waveforms are very similar in MEC and MCC. If compared with the results of the previous section, the mean peak-to-peak value recorded in case of central fibre is lower than the one in case of surface fibre. This probably indicates that the short-circuiting effect of the liquid around the nerve is not negligible.



**Figure 5.17:** Potential values on each ring from a  $5.7 \mu\text{m}$  surface-located fibre, obtained different electrode types.

Effect of the electrode type on recorded potential (V) $5.7 \mu\text{m}$ fibre in $(2/3r, 0)$		
	<i>MEC</i>	<i>MCC</i>
Ring 1	1.9111e-06	1.9722e-06
Ring 2	3.132e-06	3.33e-06
Ring 3	3.2292e-06	3.3223e-06
Ring 4	2.0975e-06	2.1469e-06
Average	2.5925e-06	2.6928e-06
FWHM	0.2563	0.2570

**Table 5.15:** Peak-to-peak values, average and FWHM for the potential waveform from a  $5.7 \mu\text{m}$  surface-located fibre, obtained different electrode types.

## 5.4 Multiple axons models

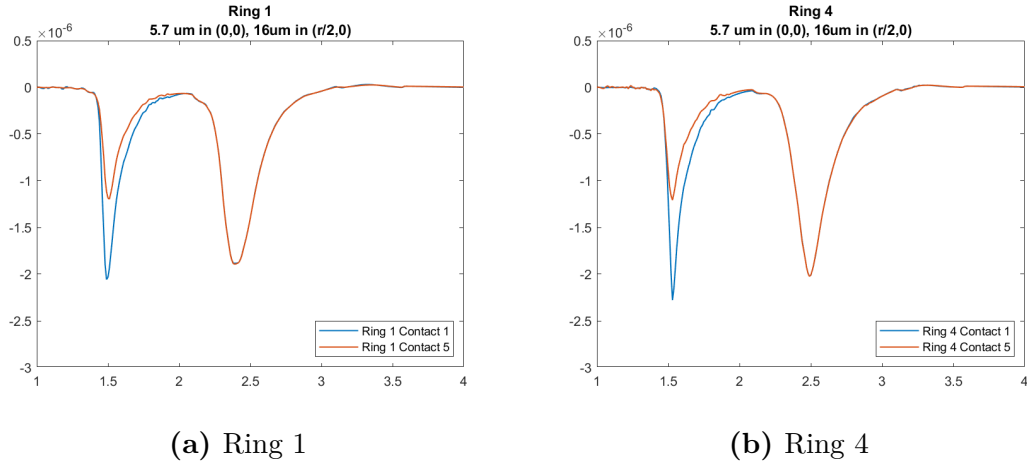
In this section, simulations with more than one fibre are presented. In the first part, simulations with just two fibres are shown, while in the second there are some examples of CAP recording based on a nerve that contains all the 9 fibre diameters that have been created through the NEURON software. The simulations presented in this section are made with the use of a MCC, to present the effect, highlighted in the previous sections, of the spatial discrimination capacity of this type of electrode.

### 5.4.1 Simulations with 2 fibres

In this section, simulations with 2 fibers are presented. The original single-fibre models are slightly complicated with the insertion of two different axons, seeing for the first time an approximation of a CAP, in a highly ideal and controlled case, namely that of two fibres with very different diameters: 16 and 5.7  $\mu\text{m}$ . This case is difficult to replicate in the real case, as the fibers are recruited in reverse order of diameter. Consequently, it would be very difficult to selectively recruit fibers with such different diameters and obtaining such a CAP. The first test presented consists of the simulation of a 5.7 together with a 16  $\mu\text{m}$  fiber. These fibers are simulated in positions (0,0) and ( $r/2$ , 0). Waveforms are observed showing a primitive CAP. The contacts of the MCC are numbered starting from the one in position ( $r_{cuff}$ , 0), counterclockwise. Consequently, in specific simulations, the first contact of each ring is the one closest to the surface fiber, while the fifth is the most distant. For this section the analysis of the amplitudes is not presented, as it is considered to be of little significance. In fact, the use of more complex algorithms would be required here in order to isolate the individual conduction velocities and attempt a first selective recording approach assisted by the algorithms, which however it is not the main focus of this work.

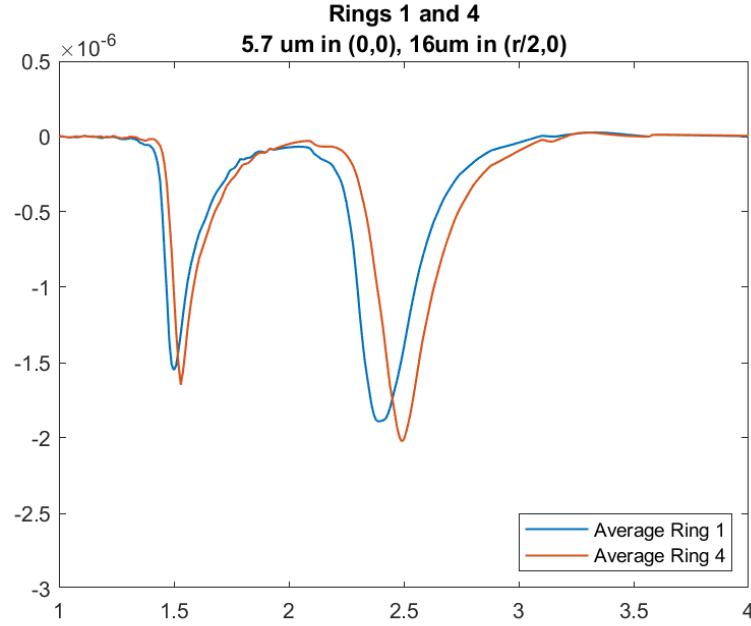
Presentation of the results is therefore limited to showing the plots of the recorded potentials and qualitatively analyzing their morphology.

### 5.7 $\mu\text{m}$ central fibre and 16 $\mu\text{m}$ superficial fibre



**Figure 5.18:** Plots on the potentials on the 1st and 5th contact of a 4-rings MCC. Nerve with central 5.7  $\mu\text{m}$  fibre and surface 16  $\mu\text{m}$  fibre.

In the figures 5.18 and 5.19 the first peak corresponding to the fastest fiber, i.e.



**Figure 5.19:** Plots on the contact-averaged potentials on the first and last ring of the MCC. Nerve with central  $5.7 \mu\text{m}$  fibre and surface  $16 \mu\text{m}$  fibre.

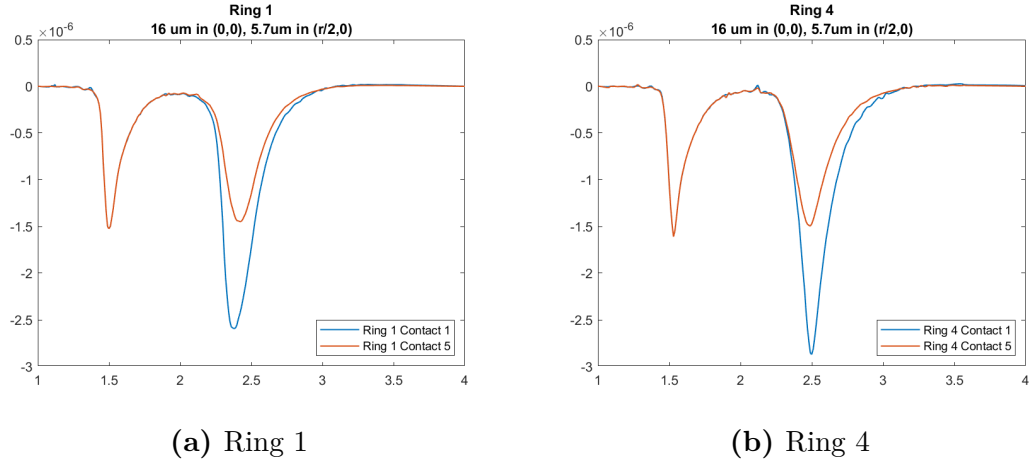
the  $16 \mu\text{m}$  fiber, and the second peak corresponding to the  $5.7 \mu\text{m}$  fiber are evident. In this case, the surface fiber is the fastest.

Figure 5.18 shows what is expected, that is, that the voltage values sensed by the more distant contact and the closest contact differ considerably in amplitude. No difference, however, is detected on the  $5.7 \mu\text{m}$  fiber.

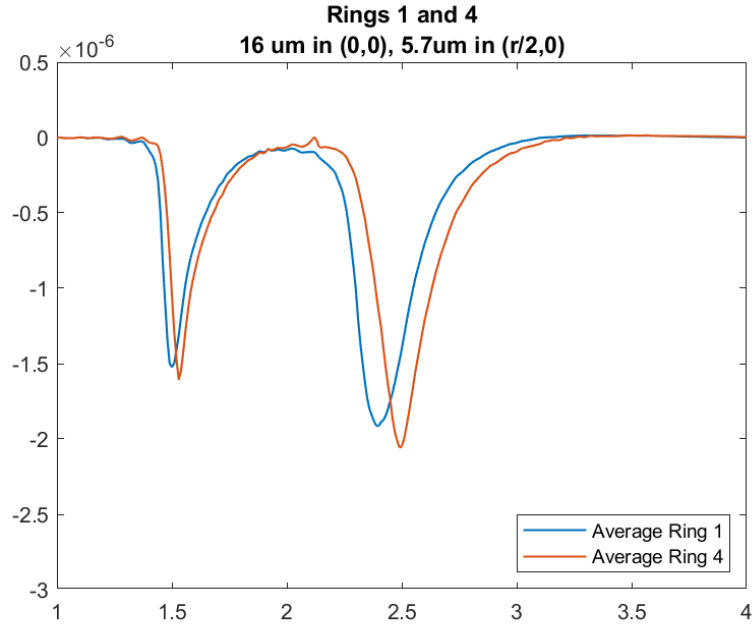
In figure 5.19, in addition to the delay of the fiber, is also possible to appreciate the faster propagation of APs in the  $16 \mu\text{m}$  fiber, which presents the peaks on the first and last ring much closer than what happens to the  $5.7 \mu\text{m}$  fiber.

### 16 $\mu\text{m}$ central fibre and 5.7 $\mu\text{m}$ superficial fibre

In the case the surface fiber is the one with the smallest diameter, the difference in the amplitudes recorded by diametrically opposite contacts is obviously found on the second peak visible in the recordings. The distance between the peaks, as well as the speed of propagation, visible in figures 5.20 and 5.21, are comparable to those of the previous experiment. What changes, consequently, is not the propagation characteristics of the waveforms, but rather the relative amplitude of the peaks, that change exchanging the fibers, as expected.



**Figure 5.20:** Plots on the potentials on the 1st and 5th contact of a 4-rings MCC. Nerve with central 5.7  $\mu\text{m}$  fibre and surface 10  $\mu\text{m}$  fibre.



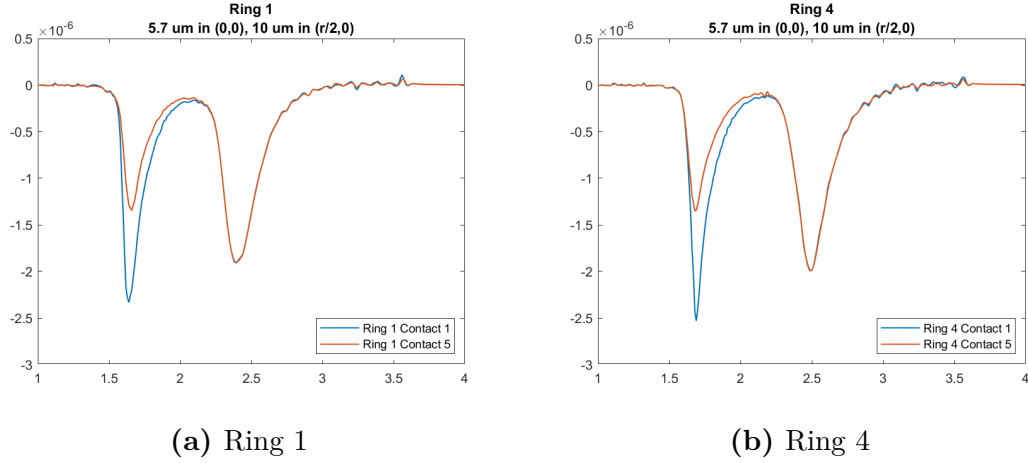
**Figure 5.21:** Plots on the contact-averaged potentials on the first and last ring of the MCC. Nerve with central 5.7  $\mu\text{m}$  fibre and surface 10  $\mu\text{m}$  fibre.

### 5.7 $\mu\text{m}$ and 10 $\mu\text{m}$ fibres

Per confronto, qui sono riportati i risultati di due esperimenti analoghi ai due precedenti, ma con fibre di diametro più ravvicinato. I risultati sono riportati

nelle figure 5.22, 5.23, 5.24 e ??.

For comparison, here are the results of two experiments similar to the previous two, but with fibers of a closer diameter, namely  $5.7 \mu\text{m}$  and  $10 \mu\text{m}$ . The results are shown in figures 5.22, 5.23, 5.24 and 5.25.



**Figure 5.22:** Plots on the potentials on the 1st and 5th contact of a 4-rings MCC. Nerve with central  $16 \mu\text{m}$  fibre and surface  $5.7 \mu\text{m}$  fibre.

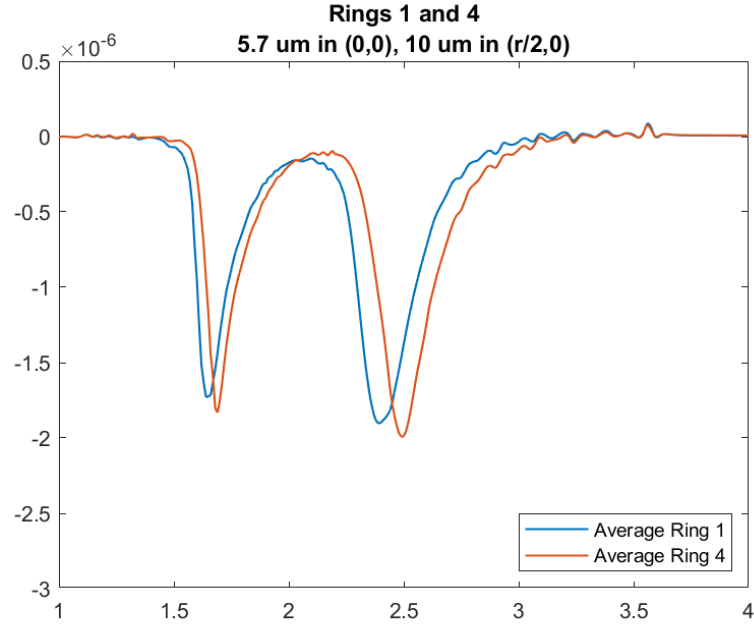
As expected, the graphs show that the peaks corresponding to the two fibers approach to each other, as the CVs are more similar.

### 5.4.2 Simulations of CAPs

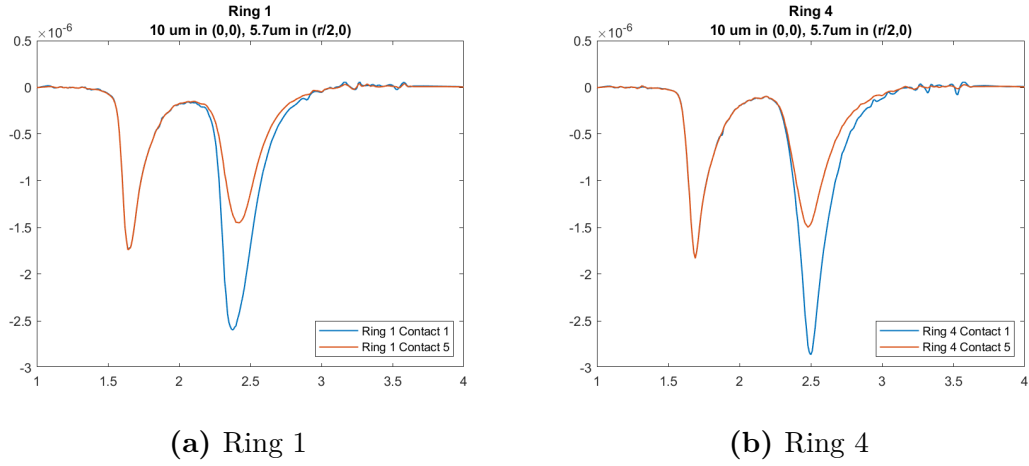
In a purely demonstrative way, a simulation performed with all the fibers obtained with NEURON is reported here. The arrangement of the fibers is the one shown in table 5.16, and in figure 5.26. The fiber diameter chosen in this case is  $2 \mu\text{m}$  and the materials assigned to the nervous tissue are frequency-dependent, to avoid the noise encountered in the case of simulations with simple tissue, given the greater complexity of the expected waveform .

The results, obtained by means of a cuff electrode with 4 rings and 8 contacts per ring, are those shown in the graphs in figure 5.27. They represent the most faithful reproduction that can be obtained, based on the work done so far, of a CAP.

The first comparison shows the differences in the sensed waveforms from each electrode. Furthermore, comparing what is recorded by the first and last electrode of the cuff, a variation of the average waveform is observed, which shows better delineated peaks in the case of the recording by the last electrode. This is due to the different speeds of propagation of the APs, which separate more, differentiating according to the CV, as they propagate along the nerve.

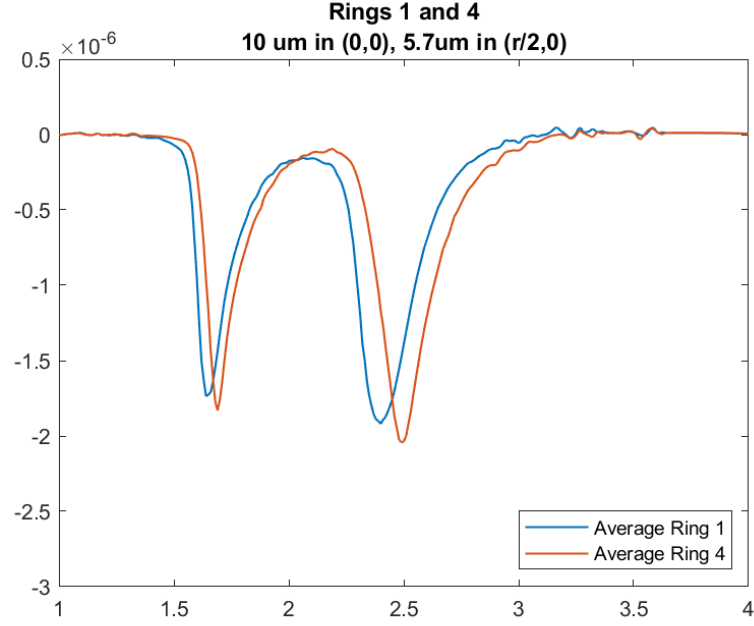


**Figure 5.23:** Plots on the contact-averaged potentials on the first and last ring of the MCC. Nerve with central  $16 \mu\text{m}$  fibre and surface  $5.7 \mu\text{m}$  fibre.



**Figure 5.24:** Plots on the potentials on the 1st and 5th contact of a 4-rings MCC. Nerve with central  $10 \mu\text{m}$  fibre and surface  $5.7 \mu\text{m}$  fibre.





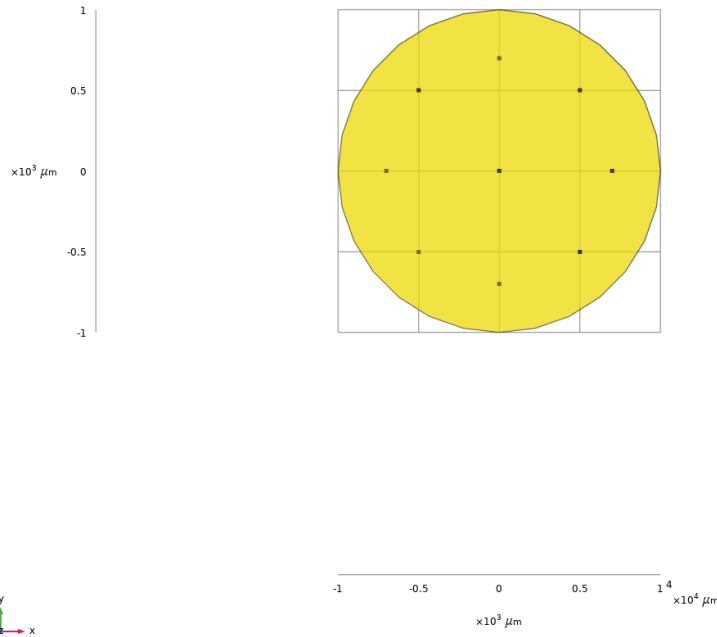
**Figure 5.25:** Plots on the contact-averaged potentials on the first and last ring of the MCC. Nerve with central 10  $\mu\text{m}$  fibre and surface 5.7  $\mu\text{m}$  fibre.

Fibre settings			
2 mm nerve model			
	<i>Diameter (<math>\mu\text{m}</math>)</i>	<i>x (<math>\mu\text{m}</math>)</i>	<i>y (<math>\mu\text{m}</math>)</i>
Fibre 1	5.7	0	0
Fibre 2	7.3	700	0
Fibre 3	8.7	500	500
Fibre 4	10	0	700
Fibre 5	11.5	-500	500
Fibre 6	12.8	-700	0
Fibre 7	14	-500	-500
Fibre 8	15	0	-700
Fibre 9	16	500	-500

**Table 5.16:** Diameters and coordinates of the fibres created for the simulation of a CAP.

## 5.5 Reliability of the results

From the analyses reported, in particular in the preliminary simulations, it is possible to get a first hint of the reliability of the model. Under experimental

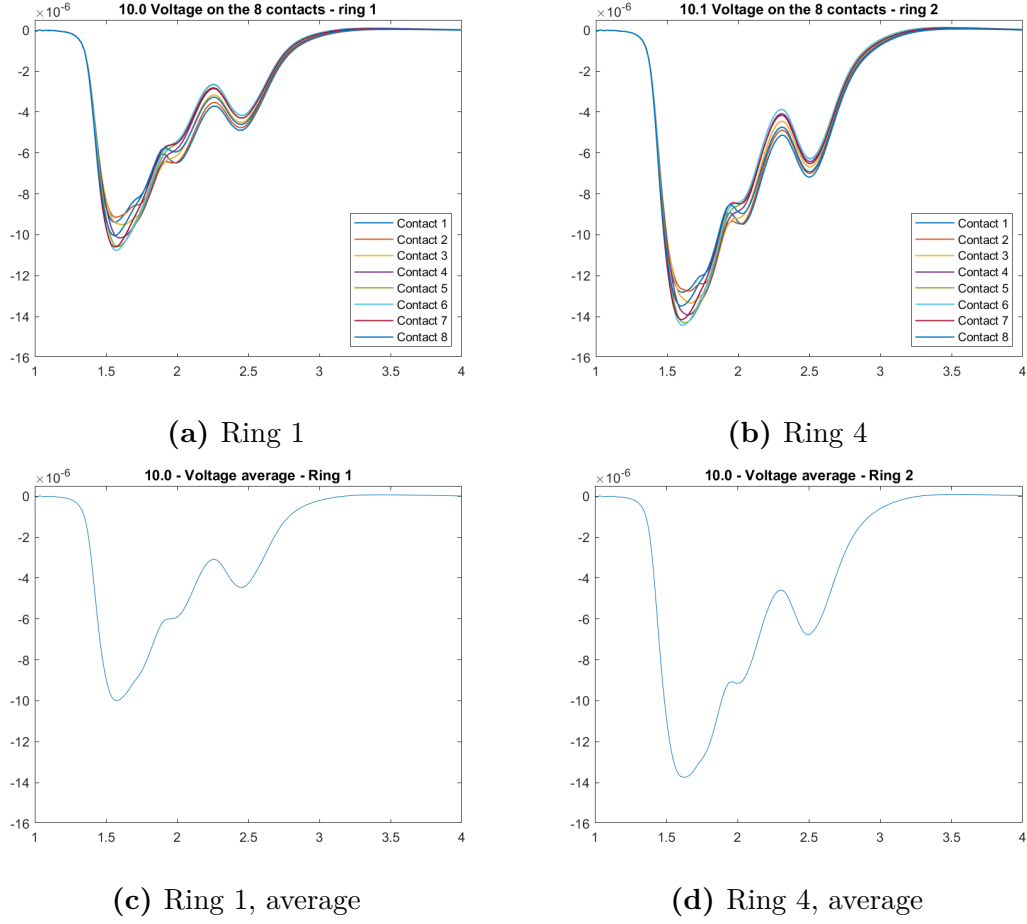


**Figure 5.26:** Section of the nerve model for CAP simulations, showing the fibre distribution.

conditions, peak values of CAPs of about 1mV are obtained from the sciatic nerve of the rat, which has a diameter of the order of 1mm, in saline solution.

On the basis of the data found in the literature and already presented in chapter 3, the total number of myelinated axons present in the sciatic nerve of the rat is about 7000-8000 (cf. table 3.2). This number includes all myelinated axons, whose diameter, as seen, can vary from 1 to 20  $\mu\text{m}$  (cf. table 2.1). By making a rough calculation, multiplying the peak recorded in the simulation with a single large-caliber fiber by the total number of axons, an overestimation of the potential obtained under experimental conditions should be obtained.

The values obtained overestimate the results of the experimental case by almost an order of magnitude. However, these very first simulations show considerable levels of simplification, due to: isotropic nervous tissue, while in the real case the transverse resistivity is much higher; lack of modeling of the highly resistive layers of perineurium, which cause a considerable attenuation of the signal; ideal condition of electrode-nerve coupling, which in the real case does not occur. To this, the most important consideration should be added, namely that this calculation is made starting from the peak value of the potential obtained from a large gauge fiber, while in reality it must be considered that lower caliber fibers provide a less significant contribution to the signal, and that the CAP spreads over time due to the different conduction velocities of the fibers, causing the peaks corresponding



**Figure 5.27:** Example of a CAP simulated within the simulation environment created.

to the contribution of the individual fibers to arrive out of phase in time and consequently not to add up to each other at their maximum amplitude.

The results obtained and commented in the specific sections are however generally in agreement with what is expected in the real case, in particular regarding the variations in amplitude and morphology of the signal experienced in the various simulation conditions. With the appropriate improvements, the model will be able to get even closer to the results obtained under experimental conditions, while already providing an excellent starting point for the experimentation of signal processing algorithms and for the construction of classifiers, in particular those based on source localization.

## Chapter 6

# Conclusions

The work presented in this thesis constitutes the start of the development of an instrument that can make it possible to obtain data that mimic the signals recorded by cuff electrodes on the surface of a nerve. The model developed so far allows the sampling of potentials similar to those electrically evoked in *ex vivo* conditions. The model has limitations due to the low number of fibres inserted so far, as well as to the approximations that were necessary to start the modelling of the nerve. However, the model shows overall good compliance with what was expected, based on the experimental results found in the state of the art of extraneural signal recording, already providing a starting point for the development of signal processing and classification algorithms. Its flexibility and the popularity of the tools used for its development leave room for extensive improvements, to make the model even closer to the original structure.

The future development envisaged for this work includes the improvement of the model features and the development of signal processing algorithms. Particularly in the short term, the model can, according to the current state, easily be complicated in order to obtain a more faithful reproduction of the nerve structure. Examples of interventions are increasing the number of axons, creating new types of fibre to be imported into the model, modifying the stimulation parameters by introducing simulation of natural nervous activity, creating a fascicular structure and in particular modelling the perineurium, which will help to lower the levels of recorded potential due to its high resistivity. Further improvements that can be made concern the definition of materials, in particular, that of the permittivity values for the conductors used and, in addition, the introduction of anisotropy.

The use of a computational model such as the one proposed in this work would save the time and resources needed, to date, to conduct exploratory experiments on animals, enabling the production of data on which to optimize the experimental conditions and deciding in advance which characteristics the electrode should have to achieve the desired results in terms of selective recording of extraneural signals.

# Appendix A

## Scripts

In this appendix, the scripts used for the generation of the final results are presented. Section A.1 lists the scripts used within the software NEURON for the generation of the membrane currents in the NOR and their export in *.txt* files. Section A.2 lists the Java code written within COMSOL for automating repetitive tasks in the creation of the model. Finally, section A.3 lists the MATLAB code written to load the output of the COMSOL simulations and for producing the plots that are shown in the chapters 4 and 5.

### A.1 NEURON

In this section, the code generated in NEURON is listed. This code is mainly a modification of the MRG axon, adapted to the need of having an higher axon length and the automatic creation of files from the simulation. Scripts are organised modularly. This means that a main script, *mosinit.hoc*, is run, having inside calls to all the scripts that need to be executed and that deal each with a different task. Examples are shown for a 16  $\mu\text{m}$  fibre.

#### A.1.1 *mosinit.hoc*

Is the initialisation code, that contains reference to all the other scripts that need to be executed.

```
// mosinit.hoc

// this is an initialisation file that:
// - loads the NEURON GUI
// - creates an MRG axon
// - opens the session with the plots
```

```
// - saves to files the currents for each node

load_file("nrngui.hoc")

xopen("MRGaxon.hoc")
xopen("vec_generationLoop.hoc")

//opens gui with original MRG axon graphs
xopen("MRGaxon.ses")

//opens .ses files
xopen("propagation.ses") //voltage propagation along the nodes
xopen("i_membrane.ses") //current propagation along the nodes

//automatic initialisation and run of the program
init()
run()

//this is for saving to a file the membrane currents
xopen("currentConversion.hoc")
xopen("fileCreationLoop.hoc")
```

### **A.1.2 *MRGaxon.hoc***

This file is a modification of the MRG axon original script by McIntyre et Al. [33]. It has been adapted to the needs of this particular simulation.

```
/*-----
2/02
Cameron C. McIntyre
SIMULATION OF PNS MYELINATED AXON
```

This model is described in detail in:

McIntyre CC, Richardson AG, and Grill WM. Modeling the excitability of mammalian nerve fibers: influence of afterpotentials on the recovery cycle. *Journal of Neurophysiology* 87:995-1006, 2002.

This model can not be used with NEURON v5.1 as errors in the extracellular mechanism of v5.1 exist related to xc. The original stimulations were run on v4.3.1. NEURON v5.2 has corrected the limitations in v5.1 and can be used to run this model.

```

-----*/
/*-----
Here the original script has been slightly modified in 2020
by Fabiana Del Bono. Main fields to be edited are commented.
-----*/

load_proc("nrnmainmenu")

proc model_globels() {
  celsius=37
  v_init=-80 //mV//
  dt=0.005 //ms//
  tstop=10
  //Intracellular stimulation parameters//
  istim=5
  delay=1
  pw=0.1
  //topological parameters//
  axonnodes=41 //EDIT
  paranodes1=2*(axonnodes-1)
  paranodes2=2*(axonnodes-1)
  axoninter=6*(axonnodes-1)
  axontotal=axonnodes+paranodes2+paranodes2+axoninter
  //morphological parameters//
  fiberD=16.0 //EDIT
  //choose from 5.7, 7.3, 8.7, 10.0, 11.5, 12.8, 14.0, 15.0, 16.0
  paralength1=3
  nodelength=1.0
  space_p1=0.002
  space_p2=0.004
  space_i=0.004
  //electrical parameters//
  rhoa=0.7e6 //Ohm-um//
  mycm=0.1 //uF/cm2/lamella membrane//
  mygm=0.001 //S/cm2/lamella membrane//
}
model_globels ()

//calculating diameter-dependant variables
proc dependent_var() {
  if (fiberD==5.7) {g=0.605 axonD=3.4 nodeD=1.9 paraD1=1.9

```

---

```

    paraD2=3.4 deltax=500 paralength2=35 nl=80}
if (fiberD==7.3) {g=0.630 axonD=4.6 nodeD=2.4 paraD1=2.4
    paraD2=4.6 deltax=750 paralength2=38 nl=100}
if (fiberD==8.7) {g=0.661 axonD=5.8 nodeD=2.8 paraD1=2.8
    paraD2=5.8 deltax=1000 paralength2=40 nl=110}
if (fiberD==10.0) {g=0.690 axonD=6.9 nodeD=3.3 paraD1=3.3
    paraD2=6.9 deltax=1150 paralength2=46 nl=120}
if (fiberD==11.5) {g=0.700 axonD=8.1 nodeD=3.7 paraD1=3.7
    paraD2=8.1 deltax=1250 paralength2=50 nl=130}
if (fiberD==12.8) {g=0.719 axonD=9.2 nodeD=4.2 paraD1=4.2
    paraD2=9.2 deltax=1350 paralength2=54 nl=135}
if (fiberD==14.0) {g=0.739 axonD=10.4 nodeD=4.7
    paraD1=4.7 paraD2=10.4 deltax=1400 paralength2=56 nl=140}
if (fiberD==15.0) {g=0.767 axonD=11.5 nodeD=5.0
    paraD1=5.0 paraD2=11.5 deltax=1450 paralength2=58 nl=145}
if (fiberD==16.0) {g=0.791 axonD=12.7 nodeD=5.5
    paraD1=5.5 paraD2=12.7 deltax=1500 paralength2=60 nl=150}
Rpn0=(rhoa*.01)/(PI*(((nodeD/2)+space_p1)^2)-((nodeD/2)^2))
Rpn1=(rhoa*.01)/(PI*(((paraD1/2)+space_p1)^2)-((paraD1/2)^2))
Rpn2=(rhoa*.01)/(PI*(((paraD2/2)+space_p2)^2)-((paraD2/2)^2))
Rpx=(rhoa*.01)/(PI*(((axonD/2)+space_i)^2)-((axonD/2)^2))
interlength=(deltax-nodelength-(2*paralength1)-(2*paralength2))/6
}
dependent_var()

objectvar stim

create node[axonnodes]
create MYSA[paranodes1], FLUT[paranodes2], STIN[axoninter]
access node[0]

//initialisation of the model
proc initialize(){
for i=0,axonnodes-1 {
node[i]{
nseg=1
diam=nodeD
L=nodelength
Ra=rhoa/10000
cm=2
insert axnode

```



```
insert extracellular xrxial=Rpn0 xg=1e10 xc=0
}
}
for i=0, paranodes1-1 {
MYSa[i]{
nseg=1
diam=fiberD
L=paralength1
Ra=rhoa*(1/(paraD1/fiberD)^2)/10000
cm=2*paraD1/fiberD
insert pas
g_pas=0.001*paraD1/fiberD
e_pas=-80
insert extracellular xrxial=Rpn1 xg=mygm/(nl*2) xc=mycm/(nl*2)
}
}
for i=0, paranodes2-1 {
FLUT[i]{
nseg=1
diam=fiberD
L=paralength2
Ra=rhoa*(1/(paraD2/fiberD)^2)/10000
cm=2*paraD2/fiberD
insert pas
g_pas=0.0001*paraD2/fiberD
e_pas=-80
insert extracellular xrxial=Rpn2 xg=mygm/(nl*2) xc=mycm/(nl*2)
}
}
for i=0, axoninter-1 {
STIN[i]{
nseg=1
diam=fiberD
L=interlength
Ra=rhoa*(1/(axonD/fiberD)^2)/10000
cm=2*axonD/fiberD
insert pas
g_pas=0.0001*axonD/fiberD
e_pas=-80
insert extracellular xrxial=Rpx xg=mygm/(nl*2) xc=mycm/(nl*2)
}
```

```
}
for i=0, axonnodes-2 {
connect MYSA[2*i](0), node[i](1)
connect FLUT[2*i](0), MYSA[2*i](1)
connect STIN[6*i](0), FLUT[2*i](1)
connect STIN[6*i+1](0), STIN[6*i](1)
connect STIN[6*i+2](0), STIN[6*i+1](1)
connect STIN[6*i+3](0), STIN[6*i+2](1)
connect STIN[6*i+4](0), STIN[6*i+3](1)
connect STIN[6*i+5](0), STIN[6*i+4](1)
connect FLUT[2*i+1](0), STIN[6*i+5](1)
connect MYSA[2*i+1](0), FLUT[2*i+1](1)
connect node[i+1](0), MYSA[2*i+1](1)
}

finitialize(v_init)
fcurrent()
}
initialize()

//intracellular stimulus
node[0] stim = new IClamp(0.5)

proc setstimparams() {
    stim.del = delay
    stim.dur = pw
    stim.amp = istim
}

setstimparams()

xpanel("Stimulus parameters")
xvalue("Stimulus Amplitude (nA)", "istim", 1, "setstimparams()", 1)
xvalue("Pulse Duration (ms)", "pw", 1, "setstimparams()", 1)
xvalue("Onset Delay (ms)", "delay", 1, "setstimparams()", 1)
xpanel(100,100)
```

### A.1.3 *Vec\_generationLoop*

Is the script to create and record, for each node, the vectors for time and currents that will be copied to files.

```
// Vec_generation Loop

//Declarations
objref tvec
strdef cstr

tvec = new Vector() // Creates time axis vector

//Creation of current vectors for each node
for i = 0, axonnodes-1 {

  sprint(cstr, "objref imem%d", i+1)
  execute(cstr)
  sprint(cstr, "imem%d = new Vector()", i+1)
  execute(cstr)
  sprint(cstr, "imem%d.record(&node[%d].i_membrane(0.5))", i+1, i)
  execute(cstr)
}

tvec.record(&t) // Records time axis vector

// Items are saved to file by executing the script 'save_vec.hoc'
```

### A.1.4 *currentConversion.hoc*

Is a short script to convert current, that is calculated as a distributed mechanism into its value for a point current source. In other words, it calculates the area of the NOR and multiplies it for the saved value, that is a current density, as explained in section 4.2.2.

```
// Converts distributed value for membrane current saved
// in the vector into its correspondant value assuming a
// point current source (PCS) and scales it to a value in
// Amperes ready to be used in COMSOL

// Calculating the surface of the node
nodesurface = nodelength*PI*fiberD
```

```
// Applying conversion to A
conv_factor = nodesurface*(1e-11)
for i = 0, axonnodes-1 {

    sprintf(cstr, "objref point_i%d", i+1)
    execute(cstr)

    // Creates converted PCS
    sprintf(cstr, "point_i%d = imem%d.mul(conv_factor)", i+1, i+1)
    execute(cstr)
}
```

### A.1.5 *fileCreationLoop.hoc*

This is the actual script that creates the *.txt* files as an output, in the current path.

```
// fileCreationLoop.hoc

// File creation for each node current

// Definition of a string containing the filename root
// (same as in COMSOL)
strdef txtFileName, txtFileNameRoot, txtFileExtension
txtFileNameRoot = "Currents_"
txtFileExtension = ".txt"

// Creation of a template for file generation
// (just object declaration - it is a trick because
// otherwise it would not be possible in a loop)
begintemplate createObj
public _Object_
objref _Object_ // Creates new object
endtemplate createObj

objref txtFile, imem
strdef imemCreation

for j = 0, axonnodes-1 {

    sprintf(imemCreation, "imem = point_i%d", j+1)
    execute(imemCreation)
```

```
// Filename string creation
sprintf(txtFileName,"%s%d%s",txtFileNameRoot,j,txtFileExtension)

// File creation
txtFile = new createObj()
txtFile = new File(txtFileName)
txtFile.wopen(txtFileName)

// File writing (for loop)
for i = 0,tvec.size()-1 {
txtFile.printf("%f\t %e\n",tvec.x[i],imem.x[i])
}

// File closing
txtFile.close()
}
```

## A.2 COMSOL

The COMSOL code listed here, written in Java within the Application builder, has been used for defining the axons to be created inside the nerve structure. These scripts are executed sequentially. The last script is the one used to export the potentials recorded on the electrode contacts of a MCC.

### A.2.1 Initialisation of counters

```
1 // Setting counters to 0
2 with(model.param("par4")); // Utility parameters
3   set("Axons_counter", "0"); // Set the counter of total
   axons
4   descr("Axons_counter", "Counter of total axons");
5   endwith();
6
7 // Iterate i from 0 to N minus 1
8 for (int i = 1; i <= 9; ++i) {
9   // Repeated N times
10  with(model.param("par4")); // Utility parameters
```

```

11     set("Count_"+i, "0"); //set the counter of axons
    having current diameter
12     descr("Count_"+i, "Counter of axons having current
    diameter");
13     endwith();
14 }
15 // Setting Utility parameters

```

### A.2.2 Import currents from Neuron

```

1 // This method is for importing the currents stored in
    the files produced by NEURON
2 // Imports data about currents from .txt files
3 // Data are stored as interpolation functions
4
5 // Reading from the defined parameters
6
7 int axons = 9; // Setting number of axons
8
9 // Setting the parameters for reading the files
10 String name = "Currents_"; // Filename (numeration
    excluded)
11 String ext = ".txt"; // File extension
12
13 // Loop over axons (first axon has number 1)
14 for (int j = 1; j <= axons; ++j) {
15
16     String a = Integer.toString(j);
17     double diam = model.param().evaluate("Diam_"+a);
18     String d = toString(diam);
19
20     String path = "C:\\Users\\fabis\\Desktop\\Axon
        Models\\Simulations with all diameters\\Built-in
        current conversion - e format\\"+d+"\\"; // File path
21
22     // Create group of currents
23     model.nodeGroup().create("grpc"+a, "GlobalDefinitions");
24     with(model.nodeGroup("grpc"+a));
25     set("type", "func");

```

```

26 endwhile();
27 model.nodeGroup("grpc"+a).label("Currents - Fibre "+a);
   // Rename group
28
29 int N = (int)
   Math.round(model.param().evaluate("Node_number_"+a));
   // Setting the number of nodes, reading from the
   parameters
30
31 // Loop over all the nodes in the current axon for
   creating current functions
32 // Iterate i from 0 to N minus 1
33 for (int i = 0; i < N; ++i) {
34     // Repeated N times
35     String x = Integer.toString(i);
36     model.func().create("int_"+a+"_"+x, "Interpolation");
   // Creates new interpolation function
37     model.func("int_"+a+"_"+x).label("Current - Fibre
   "+a+" Node "+x); // Insert label
38     with(model.func("int_"+a+"_"+x));
39         set("source", "file");
40         set("filename", path+name+x+ext);
41     endwhile();
42     model.func("int_"+a+"_"+x).importData();
43     with(model.func("int_"+a+"_"+x));
44         set("argunit", "ms");
45         set("fununit", "A");
46     endwhile();
47
48     // Add to group
49     model.nodeGroup("grpc"+a).add("func", "int_"+a+"_"+x);
50 } // End of loop over nodes in current axon
51 } // End of loop over axons

```

### A.2.3 Creation of the axons

```

1 // Executes the sequence of scripts that is necessary for
   creating a new axon
2 A_Axon_settings();

```

```

3 B_fiber_properties();
4 C_FINAL_point_creation();

```

***A\_Axon\_settings()*** - Set location and diameter of the new axon

```

1 // Getting values from parameters for the new axon and
  // writing them in the lists
2
3 // Getting valules for new axon
4 //double diam = model.param().evaluate("New_diameter");
5 //double x = model.param().evaluate("New_x");
6 //double y = model.param().evaluate("New_y");
7 // Setting Utility parameters
8
9 double axon = model.param().evaluate("Axons_counter");
10 axon++;
11 with(model.param("par4")); // Utility parameters
12   set("Axons_counter", toString(axon)); // Set the
    // counter of total axons
13   descr("Axons_counter", "Counter of total axons");
14 endwhile();
15
16 String i =
    toString(model.param().evaluate("Axons_counter"), 0);
17
18 // Writing the values in the sum-up tables
19 with(model.param("par8")); // Diameter list
20   set("diam_"+i,
    toString(model.param().evaluate("New_diameter"), 1));
21   descr("diam_"+i, "Diameter of axon n. "+i);
22 endwhile();
23
24 with(model.param("par6")); // x coord list
25   set("x_"+i, toString(model.param().evaluate("New_x"),
    0));
26   descr("x_"+i, "x coord of axon n. "+i);
27 endwhile();
28
29 with(model.param("par7")); // y coord list

```



```
30 set("y_"+i, toString(model.param().evaluate("New_y"),  
    0));  
31 descr("y_"+i, "y coord of axon n. "+i);  
32 endwith();
```

### ***B\_fiber\_properties()*** - Set fibre properties

```
1 // Reading current diameter  
2 double diam = model.param().evaluate("New_diameter");  
3 int fiberType = 0;  
4  
5 if (diam == 5.7) {  
6     // condition was true  
7     fiberType = 1;  
8 }  
9 else if (diam == 7.3) {  
10    // condition was true  
11    fiberType = 2;  
12 }  
13 else if (diam == 8.7) {  
14    // condition was true  
15    fiberType = 3;  
16 }  
17 else if (diam == 10) {  
18    // condition was true  
19    fiberType = 4;  
20 }  
21 else if (diam == 11.5) {  
22    // condition was true  
23    fiberType = 5;  
24 }  
25 else if (diam == 12.8) {  
26    // condition was true  
27    fiberType = 6;  
28 }  
29 else if (diam == 14) {  
30    // condition was true  
31    fiberType = 7;  
32 }
```

```

33 else if (diam == 15) {
34     // condition was true
35     fiberType = 8;
36 }
37 else if (diam == 16) {
38     // condition was true
39     fiberType = 9;
40 }
41
42 // Getting the number of axons with this diameter
43 String t = Integer.toString(fiberType);
44 double count = model.param().evaluate("Count_"+t);
45 count++;
46
47 // Setting Utility parameters
48 with(model.param("par4")); // Utility parameters
49     set("Current_fibre_type", toString(fiberType)); //Set
        current fibre type
50     descr("Current_fibre_type", "Number correspondent to
        the fiber diameter in the parameters tables");
51     set("fiberD", toString(diam)); //set current fibre
        diameter
52     descr("fiberD", "Current fiber diameter");
53     set("Count_"+t, toString(count)); //set the counter of
        axons having current diameter
54     descr("Count_"+t, "Counter of axons having current
        diameter");
55 endwhile();
56
57 // Getting the total number of axons created
58
59 double axcount = model.param().evaluate("Axons_counter");
60 String x = toString(axcount);
61
62 with(model.param("par18")); // Fiber type list
63     set("Axon_"+x, t);
64 endwhile();

```

\_\_FINAL\_point\_creation - Create points

```

1 // Inserts points along the axon
2
3 int fibertype = (int)
    Math.round(model.param().evaluate("Current_fibre_
    type")); // Setting the number of nodes, reading from
    the parameters
4
5 // Lines for adapting parameters to the chosen fiber type
6 String s = Integer.toString(fibertype);
7 double deltax = model.param().evaluate("Interlength_"+s);
    // Setting internodal distance for the specific fiber
    type
8 int N = (int)
    Math.round(model.param().evaluate("Node_number_"+s));
    // Number of nodes in current axon fiber type
9
10 double space = 0; //initialisation to this value
11
12 // Defining current axon
13 int axon = (int)
    Math.round(model.param().evaluate("Axons_counter"));
14 String a = Integer.toString(axon);
15
16 // Defining coordinates
17 double xcoord = model.param().evaluate("New_x"); //
    Xcoordinate of the axon
18 double ycoord = model.param().evaluate("New_y"); //
    Ycoordinate of the axon
19
20 // Iterate i from 0 to N minus 1
21 for (int i = 0; i < N; ++i) {
22     // Repeated N times
23     String x = Integer.toString(i);
24     model.component("comp1").geom("geom1").create("ax"+a+"_
    pt"+x, "Point"); // name is of type "ax1_pt2"
25     with(model.component("comp1").geom("geom1").feature("
    ax"+a+"_pt"+x));
26     setIndex("p", xcoord, 0);
27     setIndex("p", ycoord, 1);
28     setIndex("p", space, 2);

```

```

29 endwhile();
30 model.component("comp1").geom("geom1").run("ax"+a+"_pt
    "+x);
31 space = space+deltax;
32
33 // Create label "Axon x - node y"
34 model.component("comp1").geom("geom1").feature("ax"+a+
    "_pt"+x).label("Axon "+a+" - node "+x);
35
36 // Enables selection of point in physics
37 with(model.component("comp1").geom("geom1").feature("ax
    "+a+"_pt"+x));
38     set("selresult", true);
39 endwhile();
40 }

```

## A.2.4 Create electrode selection

```

1 Intersection
2
3 int nring = 4;
4 int ncont = 8;
5 int count = 0; int selcont;
6 // Iterate i from 0 to N minus 1
7 for (int i = 1; i <= nring; ++i) {
8     // Repeated N times
9     // Iterate i from 0 to N minus 1
10    for (int j = 1; j <= ncont; ++j) {
11        // Repeated N times
12        count = count+1;
13        selcont = nring+j; // Because the tag for contacts
14        starts from number of rings+1
15        String is = Integer.toString(i);
16        String js = Integer.toString(j);
17        String counts = Integer.toString(count);
18        String selconts = Integer.toString(selcont);
19        model.component("comp1").selection().create("int"+
20        counts, "Intersection");

```

```

19     model.component("comp1").selection("int"+counts).
    label(is+"_"+js);
20
21     with(model.component("comp1").selection("int"+counts));
    set("input", new String[]{"sel"+i, "sel"+selconts});
22     endwith();
23
24 }
25 }

```

### A.2.5 Insert point current sources

```

1 // Sets point current sources in the defined points -
    still some small problems to be solved
2
3 int axons = (int)
    model.param().evaluate("Axons_counter"); // Setting
    number of axons
4
5 // Loop over axons (first axon has number 1)
6 for (int j = 1; j <= axons; ++j) {
7
8     String a = Integer.toString(j);
9
10    // Create group of point current sources
11    model.nodeGroup().create("grpp"+a, "Physics", "ec");
12    model.nodeGroup("grpp"+a).label("Axon"+a);
13
14    // Getting current fiber type
15    double fiberType = model.param().evaluate("Axon_"+a);
16    String t = toString(fiberType);
17    // Getting the number of nodes to loop on
18    int N = (int) model.param().evaluate("Node_number_"+t);
19
20    // Loop over all the nodes in the current axon for
    creating current functions
21    // Iterate i from 0 to N minus 1
22    for (int i = 0; i < N; ++i) {
23        // Repeated N times

```

```

24     String x = Integer.toString(i);
25
26     model.component("comp1").physics("ec").create("pcs_"+a+
27     "_" + x, "PointCurrentSource", 0);
28     with(model.component("comp1").physics("ec").feature(
29     "pcs_"+a+ "_" + x));
30     set("Qjp", "int_"+a+ "_" + x + "(t)*Conv_factor");
31     endwith();
32
33     // Assigning the source to a point
34     model.component("comp1").physics("ec").feature("pcs_"
35     +a+ "_" + x).selection().named("geom1_ax"+a+ "_pt"+x+
36     "_pnt");
37
38     // Add to group
39     model.nodeGroup("grpp"+a).add("pcs_"+a+ "_" + x);
40
41 } // End of loop over nodes in current axon
42 } // End of loop over axons

```

## A.2.6 Export voltage on the contacts

```

1 Vol_average_creation
2
3 // Creates average voltage on single contacts and average
4 // on rings
5
6 String simulationCode = "2.2.5";
7 String path = "C:\\Users\\fabis\\Desktop\\COMSOL\\Voltage
8 reading
9 simulation\\Exported_voltage\\" + simulationCode + "\\";
10
11 int nring = 4;
12 int ncont = 8;
13 int count = 0; int selcont;
14 // Iterate i from 0 to N minus 1
15 for (int i = 1; i <= nring; ++i) {
16     // Repeated N times
17     // Iterate i from 0 to N minus 1
18     for (int j = 1; j <= ncont; ++j) {

```

```

15 // Repeated N times
16
17 count = count+1;
18 selcont = nring+j; // Because the tag for contacts
starts from number of rings+1
19 String is = Integer.toString(i);
20 String js = Integer.toString(j);
21 String counts = Integer.toString(count);
22 String selconts = Integer.toString(selcont);
23
24
25 // Create new volume average
26 model.result().numerical().create("av"+counts,
"AvVolume");
27 with(model.result().numerical("av"+counts)); //
Select electric potential
28 set("expr", new String[]{"V"});
29 set("descr", new String[]{"Electric potential"});
30 endwith();
31
32 // Select entities
33 model.result().numerical("av"+counts).selection().
named("int"+counts); // Selection of the contact
34 model.result().table().create("tbl"+counts, "Table");
// Creation of the table
35 model.result().table("tbl"+counts).comments("Volume
Average "+counts+" "+is+"_"+js+" {av"+counts+"}"); //
Info
36 with(model.result().numerical("av"+counts));
37 set("table", "tbl"+counts); // Set table to contain
selected volume average
38 endwith();
39 model.result().numerical("av"+counts).setResult(); //
Set the values in the table
40 model.result().table("tbl"+counts).label(is+"_"+js);
// Name the table
41 with(model.result().table("tbl"+counts)); // Store on
file
42 set("storetable", "onfile");
43 set("filename", path+is+"_"+js+".txt");
44 endwith();

```

```

45     model.result().table("tbl"+counts).label("Voltage_"+is+
46         "_"+js); // Create meaningful label name for the table
47         in COMSOL
48
49     }
50 }
51
52 // Creation of files with average on the ring
53 for (int i = 1; i <= nring; ++i) {
54     count = count+1;
55     String counts = Integer.toString(count);
56     String is = Integer.toString(i);
57
58     model.result().numerical().create("av"+counts,
59         "AvVolume");
60     with(model.result().numerical("av"+counts));
61         set("expr", new String[]{"V"});
62         set("descr", new String[]{"Electric potential"});
63     endwhile();
64     model.result().numerical("av"+counts).selection().named
65         ("sel"+is);
66     model.result().table().create("tbl"+counts, "Table");
67     model.result().table("tbl"+counts).comments("Volume
68         Average "+counts+" {av"+counts+"}");
69     with(model.result().numerical("av"+counts));
70         set("table", "tbl"+counts);
71     endwhile();
72     model.result().numerical("av"+counts).setResult();
73     model.result().numerical("av"+counts).label("Ring
74         Volume Average "+is);
75     model.result().table("tbl"+counts).label("Ring_voltage_
76         "+is);
77     with(model.result().table("tbl"+counts));
78         set("storetable", "onfile");
79         set("filename", path+"Ring_average_"+is+".txt");
80     endwhile();
81 }

```



## A.3 MATLAB

### A.3.1 NOR current plots

```

1 %% Script for reading current values automatically from .
  txt file
2 % For each script (corresponding to each node) a matrix is
  created. All
3 % matrices are stored in a cell.
4
5 close all
6 clear all
7 clc
8
9 %% Parameters
10 N = 41; % Number of NOR
11 filename = 'Currents_%d.txt';
12 c = cell(N,1); % Cell to store current data
13
14 %% Loop for reading files and saving them to matrix stored
  in cell array
15 for i = 1:N
16     c{i,1} = readmatrix(sprintf(filename, i-1));
17 end % i-1 because neuron starts counting from 0
18
19 %% Plots
20 fig = 1; % Figure number
21
22 % Plots all the current plots superimposed
23 for i = 1:N
24
25     figure(fig)
26     plot(c{i,1}(:,1),c{i,1}(:,2),'DisplayName',sprintf
  ('NOR %d',i-1))
27
28     title('NOR Currents')
29     hold on
30
31 end
32

```

```

33 hold off
34
35 % Setting axis and legend
36 axis([0 3 -40 15])
37 legend('Location','southeast')
38 lgd = legend;
39 lgd.NumColumns = 4

```

### A.3.2 Voltage plots for MEC

```

1 %% Script for reading voltage values automatically from .
  txt file
2 close all
3 clear all
4 clc
5 format short
6 %%
7 sim = 6.4;
8
9 %% Parameters
10 N = 4; % Number of rings
11 Nc = 1; % Number of contacts per ring
12 filename = 'ring%d.txt';
13 c = cell(N,Nc); % Cell to store voltage data
14
15 %% Loop for reading files and saving them to matrix stored
  in cell array
16
17 for i = 1:N
18     for j = 1:Nc
19         c{i,j} = readmatrix(sprintf(filename, i));
20     end
21 end
22
23 %% Plots
24 % Optimised for ring electrodes
25
26 fig = 1; % Figure number
27 x = N; % Counter for subplot

```

```

28
29 % Plot first row with all signal superposed
30 for i = 1:N
31     for j = 1:Nc
32         figure(fig)
33         subplot(Nc+1,N,[1:N])
34         plot(c{i,j}(:,1),c{i,j}(:,size(c{i,j},2)), '
35             title('All-rings voltage plot')
36         hold on
37     end
38 end
39 x = x+1;
40 hold off
41
42 for i = 1:N
43     for j = 1:Nc
44         h = figure(fig+100)
45         plot(c{i,j}(:,1),c{i,j}(:,size(c{i,j},2)), '
46             DisplayName',sprintf('Ring %d',i),'linewidth',1)
47         title(sprintf('%.1f - All-rings voltage plot',sim)
48         )
49         hold on
50     end
51 end
52 axis tight
53 legend('Location','southeast')
54 saveas(h,sprintf('%.1f V all.png',sim))
55 hold off
56
57 % Plot single graphs
58 for i = 1:N
59     for j = 1:Nc
60         figure(fig)
61         subplot(Nc+1,N,x)
62         plot(c{i,j}(:,1),c{i,j}(:,size(c{i,j},2)), '
63             title(sprintf('Voltage - Ring %d',i))
64         x = x+1;
65     end
66

```

```

67
68 end
69
70 for i = 1:N
71     for j = 1:Nc
72
73         h = figure(fig+100)
74         plot(c{i,j}(:,1),c{i,j}(:,size(c{i,j},2)))
75         title(sprintf('%.1f - Voltage - Ring %d',sim,i))
76
77         axis tight
78         saveas(h,sprintf('%.1f V Ring %d.png',sim,i))
79
80     end
81
82 end
83
84 %% Calculating Amplitude for each channel
85 for i = 1:N
86     for j = 1:Nc
87
88         A_pp(i,j) = max(c{i,j}(:,size(c{i,j},2)))-min(c{i,
89             j}(:,size(c{i,j},2)));
90     end
91 end
92
93 save(sprintf('%.1f A_pp.mat',sim),'A_pp')
94 dlmwrite(sprintf('%.1f A_pp.txt',sim),A_pp)
95 A_pp_mean = mean(A_pp,'all')
96 dlmwrite(sprintf('%.1f A_pp_mean.txt',sim),A_pp_mean)
97
98 close all

```

### A.3.3 Voltage plots for MCC

```

1 %% Script for reading voltage values automatically from .
  txt file
2 % Optimised for MCC

```

```

3 close all
4 clear all
5 clc
6 %%
7 sim = 8.6;
8 y_min = -3.5e-6;
9 y_max = 0.5e-6;
10 %% Parameters
11 Nrings = 4; % Number of rings
12 Ncontacts = 8; % Number of contacts per ring
13 filename1 = '%d_%d.txt';
14 filename2 = 'Ring_average_%d.txt';
15 c = cell(Nrings,Ncontacts); % Cell to store contact
    voltage data
16 r = cell(Nrings,1); % Cell to save average on rings
17
18 %% Loop for reading files and saving them to matrix stored
    in cell array
19
20 for i = 1:Nrings
21     for j = 1:Ncontacts
22         c{i,j} = readmatrix(sprintf(filename1, i,j));
23     end
24 end
25
26 %% Plot single graphs
27
28 fig = 1; % Figure number
29 x = 1; % Counter for subplot
30
31 % Plot single graphs
32 for j = 1:Ncontacts
33     for i = 1:Nrings
34         figure(fig)
35         subplot(Ncontacts,Nrings,x)
36         plot(c{i,j}(:,1),c{i,j}(:,size(c{i,j},2)))
37         title(sprintf('Voltage - Ring %d contact %d',i,j))
38         hold on
39         x = x+1;
40         axis([1 4 y_min y_max])
41         %save to file single windows

```

```

42         h = figure(fig+100)
43         plot(c{i,j}(:,1),c{i,j}(:,size(c{i,j},2)), '
44         DisplayName',sprintf('Ring %d Contact %d',i,j))
45         title(sprintf('%.1f - Voltage - Ring %d contact %d
46         ',sim,i,j))
47         axis([1 4 y_min y_max])
48         legend('Location','southeast')
49 %         saveas(h,sprintf('V %d_%d.png',i,j));
50     end
51 end
52 %% Plot graphs of rings superposed per contact number
53 x = 1;
54 fig = 2;
55 for j = 1:Ncontacts
56     for i = 1:Nrings
57         figure(fig)
58         subplot(Ncontacts,1,j)
59         plot(c{i,j}(:,1),c{i,j}(:,size(c{i,j},2)))
60         title(sprintf('Voltage on the %d rings - contacts
61         in position %d',Nrings,j))
62         hold on
63         x = x+1;
64         axis([1 4 y_min y_max])
65     end
66 end
67 for j = 1:Ncontacts
68     for i = 1:Nrings
69         h = figure(fig+10)
70         plot(c{i,j}(:,1),c{i,j}(:,size(c{i,j},2)), '
71         DisplayName',sprintf('Ring %d',i))
72         title(sprintf('%.1f - Voltage on the %d rings -
73         contacts in position %d',sim,Nrings,j))
74         hold on
75         axis([1 4 y_min y_max])
76     end
77     legend('Location','southeast')

```

```

78     saveas(h, sprintf('%.1f Contact %d.png', sim, j));
79     hold off
80 end
81
82
83 %% Plot graphs of rings superposed per ring
84 x = 1;
85 fig = 3;
86
87 for i = 1:Nrings
88     for j = 1:Ncontacts
89         figure(fig)
90         subplot(1, Nrings, i)
91         plot(c{i, j}(:, 1), c{i, j}(:, size(c{i, j}, 2)), '
92             title(sprintf('%.1f - Voltage on the %d contacts -
ring %d', sim, Ncontacts, i))
93             hold on
94             x = x+1;
95             axis([1 4 y_min y_max])
96     end
97 end
98
99 for i = 1:Nrings
100     for j = 1:Ncontacts
101         h = figure(fig+100)
102         plot(c{i, j}(:, 1), c{i, j}(:, size(c{i, j}, 2)), '
DisplayName', sprintf('Contact %d', j))
103         title(sprintf('%.1f Voltage on the %d contacts -
ring %d', sim, Ncontacts, i))
104         hold on
105         axis([1 4 y_min y_max])
106     end
107     legend('Location', 'southeast')
108
109     saveas(h, sprintf('%.1f Ring %d.png', sim, i));
110     hold off
111
112
113 end
114
115 %% Plot graphs with volume average

```

```
116 x = Nrings;
117 fig = 4;
118
119 for i = 1:Nrings
120
121     r{i,1} = readmatrix(sprintf(filename2, i));
122
123 end
124
125 % Plot first row with all signal superposed
126 for i = 1:Nrings
127
128     figure(fig)
129     subplot(2,Nrings,[1:Nrings])
130     plot(r{i,1}(:,1),r{i,1}(:,size(r{i,1},2)))
131     title('All-rings average voltage plot')
132     hold on
133     axis([1 4 y_min y_max])
134
135 end
136 x = x+1;
137
138 for i = 1:Nrings
139
140     h = figure(fig+100)
141     plot(r{i,1}(:,1),r{i,1}(:,size(r{i,1},2)),'DisplayName
142     ',sprintf('Ring %d',i))
143     title(sprintf('%.1f - All-rings average voltage plot',
144     sim))
145     hold on
146     axis([1 4 y_min y_max])
147
148 end
149 legend('Location','southeast')
150 saveas(h,sprintf('%.1f Average.png',sim))
151 hold off
152
153 % Plot single graphs
154 for i = 1:Nrings
155
156     figure(fig)
```



```

155     subplot(2,Nrings,x)
156     plot(r{i,1}(:,1),r{i,1}(:,size(r{i,1},2)))
157     title(sprintf('Voltage average - Ring %d',i))
158     x = x+1;
159     hold on
160         axis([1 4 y_min y_max])
161 end
162
163 for i = 1:Nrings
164
165     h = figure(fig+100)
166     plot(r{i,1}(:,1),r{i,1}(:,size(r{i,1},2)))
167     title(sprintf('%.1f - Voltage average - Ring %d',sim,i
168 ))
169     hold on
170         axis([1 4 y_min y_max])
171         saveas(h,sprintf('%.1f Average R %d.png',sim,i))
172     hold off
173 end

```

### A.3.4 Computation of FWHM

```

1 close all
2 clear all
3 clc
4 format short
5 %%
6 sim = 6.1;
7
8 %% Parameters
9 N = 4; % Number of rings
10 Nc = 1; % Number of contacts per ring
11 filename = 'Ring_average_%d.txt';
12 c = cell(N,Nc); % Cell to store voltage data
13
14 %% Loop for reading files and saving them to matrix stored
    in cell array
15
16 for i = 1:N

```

```

17     for j = 1:Nc
18         c{i,j} = readmatrix(sprintf(filename, i));
19     end
20 end
21
22 %% Calculating mean signal
23 % Plot single graphs
24 for i = 1:N
25     for j = 1:Nc
26
27         A(:,i)=c{i,j}(:,size(c{i,j},2))
28
29     end
30
31 end
32 t = interp(c{1,1}(:,1),100); %Time vector
33 mean_signal = interp(mean(A'),'100);
34 %% Calcolo fwhm
35 data = -mean_signal;
36 % Find the half max value.
37 halfMax = (min(data) + max(data)) / 2;
38 % Find where the data first drops below half the max.
39 index1 = find(data >= halfMax, 1, 'first');
40 % Find where the data last rises above half the max.
41 index2 = find(data >= halfMax, 1, 'last');
42 fwhm = index2-index1 + 1; % FWHM in indexes.
43 % OR, if you have an x vector
44 fwhmt = t(index2) - t(index1)
45
46 figure
47 plot(t,mean_signal,t,-ones(length(t),1)*halfMax)
48
49 save(sprintf('%.1f mean_signal.mat',sim),'mean_signal')

```

# Bibliography

- [1] Hannes P. Saal and Sliman J. Bensmaia. «Biomimetic approaches to bionic touch through a peripheral nerve interface». In: *Neuropsychologia* 79 (2015), pp. 344–353. ISSN: 18733514. DOI: 10.1016/j.neuropsychologia.2015.06.010. URL: <http://dx.doi.org/10.1016/j.neuropsychologia.2015.06.010> (cit. on p. 2).
- [2] Christopher E. Larson and Ellis Meng. «A review for the peripheral nerve interface designer». In: *Journal of Neuroscience Methods* 332.June 2019 (2020), p. 108523. ISSN: 1872678X. DOI: 10.1016/j.jneumeth.2019.108523. URL: <https://doi.org/10.1016/j.jneumeth.2019.108523> (cit. on pp. 2, 11–13, 17, 18).
- [3] Frank H. Netter. *The Netter collection of medical illustrations: Nervous System, Part II - Spinal cord and peripheral motor and sensory systems*. Second. Vol. 7. Saunders, Elsevier Inc., 2013 (cit. on pp. 3, 7, 8).
- [4] Cindy L. Stanfield. *Principles of Human Physiology*. 6th. University of South Alabama, Mobile, Alabama: Pearson, 2017 (cit. on pp. 4, 5).
- [5] Frank H. Netter. *The Netter collection of medical illustrations: Nervous System, Part II - Spinal cord and peripheral motor and sensory systems*. Second. Vol. 7. Saunders, Elsevier Inc., 2013 (cit. on p. 6).
- [6] Fiorenza Lotti, Federico Ranieri, Gianluca Vadalà, Loredana Zollo, and Giovanni Di Pino. «Invasive intraneural interfaces: Foreign body reaction issues». In: *Frontiers in Neuroscience* 11.SEP (2017), pp. 1–14. ISSN: 1662453X. DOI: 10.3389/fnins.2017.00497 (cit. on p. 11).
- [7] Usman Ghafoor, Sohee Kim, and Keum Shik Hong. «Selectivity and longevity of peripheral-nerve and machine interfaces: A review». In: *Frontiers in Neurobotics* 11.OCT (2017), pp. 1–21. ISSN: 16625218. DOI: 10.3389/fnbot.2017.00059 (cit. on p. 13).

- [8] Dustin J. Tyler and Dominique M. Durand. «A slowly penetrating interfascicular nerve electrode for selective activation of peripheral nerves». In: *IEEE Transactions on Rehabilitation Engineering* 5.1 (1997), pp. 51–61. ISSN: 10636528. DOI: 10.1109/86.559349 (cit. on p. 13).
- [9] Xavier Navarro, Thilo B. Krueger, Natalia Lago, Silvestro Micera, Thomas Stieglitz, and Paolo Dario. «A critical review of interfaces with the peripheral nervous system for the control of neuroprostheses and hybrid bionic systems». In: *Journal of the Peripheral Nervous System* 10.3 (2005), pp. 229–258. ISSN: 10859489. DOI: 10.1111/j.1085-9489.2005.10303.x (cit. on pp. 13–15).
- [10] Emil H. Rijnbeek, Nick Eleveld, and Wouter Olthuis. «Update on peripheral nerve electrodes for closed-loop neuroprosthetics». In: *Frontiers in Neuroscience* 12.MAY (2018), pp. 1–9. ISSN: 1662453X. DOI: 10.3389/fnins.2018.00350 (cit. on p. 13).
- [11] Breanne P. Christie, Max Freeberg, William D. Memberg, Gilles J.C. Pinault, Harry A. Hoyen, Dustin J. Tyler, and Ronald J. Triolo. «"Long-term stability of stimulating spiral nerve cuff electrodes on human peripheral nerves"». In: *Journal of NeuroEngineering and Rehabilitation* 14.1 (2017), pp. 1–12. ISSN: 17430003. DOI: 10.1186/s12984-017-0285-3 (cit. on p. 14).
- [12] G. E. Loeb and R. A. Peck. «Cuff electrodes for chronic stimulation and recording of peripheral nerve activity». In: *Journal of Neuroscience Methods* 64.1 (1996), pp. 95–103. ISSN: 01650270. DOI: 10.1016/0165-0270(95)00123-9 (cit. on pp. 15, 17).
- [13] J. Rozman, B. Zorko, and M. Bunc. «Selective recording of electroneurograms from the sciatic nerve of a dog with multi-electrode spiral cuffs». In: *Japanese Journal of Physiology* 50.5 (2000), pp. 509–514. ISSN: 0021521X. DOI: 10.2170/jjphysiol.50.509 (cit. on pp. 17, 18).
- [14] Fatemeh Karimi and Saeid R Seydnejad. «Using a Space-Time Electrode Array». In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 23.5 (2015), pp. 837–848. DOI: 10.1109/TNSRE.2014.2379594 (cit. on p. 17).
- [15] Mesut Sahin and Dominique M Durand. «Improved Nerve Cuff Electrode Recordings with Subthreshold Anodic Currents». In: *IEEE Transactions on Biomedical Engineering* 45.8 (1998), pp. 1044–1050 (cit. on p. 17).
- [16] Martin Schuettler, Vipin Seetohul, Nico J. M. Rijkhoff, Flemming V. Moeller, Nick Donaldson, and John Taylor. «Fibre-Selective Recording from Peripheral Nerves using a Multiple-Contact Cuff: Report on Pilot Pig Experiments Martin». In: *Applied Physics Letters* 99 (2011) (cit. on p. 17).

- [17] Johannes J. Struijk, Morten K. Haugland, and Morten Thomsen. «Fascicle selective recording with a nerve cuff electrode». In: *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings* 1 (1996), pp. 361–362. ISSN: 05891019. DOI: 10.1109/iembs.1996.656993 (cit. on p. 17).
- [18] José Zariffa, Mary K. Nagai, Zafiris J. Daskalakis, and Milos R. Popovic. «Influence of the number and location of recording contacts on the selectivity of a nerve cuff electrode». In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 17.5 (2009), pp. 420–427. ISSN: 15344320. DOI: 10.1109/TNSRE.2009.2023301 (cit. on p. 17).
- [19] Emma Brunton, Christoph Blau, Carolina Silveira, and Kianoush Nazarpour. «Identification of sensory information in mixed nerves using multi-channel cuff electrodes for closed loop neural prostheses». In: *International IEEE/EMBS Conference on Neural Engineering, NER* (2017), pp. 391–394. ISSN: 19483554. DOI: 10.1109/NER.2017.8008372 (cit. on p. 17).
- [20] K. Yoshida, G. A.M. Kurstjens, and K. Hennings. «Experimental validation of the nerve conduction velocity selective recording technique using a multi-contact cuff electrode». In: *Medical Engineering and Physics* 31.10 (2009), pp. 1261–1270. ISSN: 13504533. DOI: 10.1016/j.medengphy.2009.08.005 (cit. on p. 19).
- [21] B. W. Metcalfe, D. J. Chew, C. T. Clarke, N. de N. Donaldson, and J. T. Taylor. *A new method for spike extraction using velocity selective recording demonstrated with physiological ENG in Rat*. 2015. DOI: 10.1016/j.jneumeth.2015.05.003 (cit. on p. 19).
- [22] John Taylor, Benjamin Metcalfe, Chris Clarke, Daniel Chew, Thomas Nielsen, and Nick Donaldson. «A summary of current and new methods in velocity selective recording (VSR) of electroneurogram (ENG)». In: *Proceedings of IEEE Computer Society Annual Symposium on VLSI, ISVLSI* 07-10-July.July 2015 (2015), pp. 221–226. ISSN: 21593477. DOI: 10.1109/ISVLSI.2015.34 (cit. on p. 19).
- [23] David Weinstein, Leonid Zhukov, and Chris Johnson. «Lead-field Bases for Electroencephalography Source Imaging». In: *Annals of Biomedical Engineering* 28 (Sept. 2000), pp. 1059–1065. DOI: 10.1114/1.1310220 (cit. on p. 20).
- [24] José Zariffa, Mary Nagai, Martin Schuettler, Thomas Stieglitz, Zafiris Daskalakis, and Milos Popovic. «Use of an Experimentally Derived Leadfield in the Peripheral Nerve Pathway Discrimination Problem». In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 19 (May 2011), pp. 147–156. DOI: 10.1109/TNSRE.2010.2091429 (cit. on p. 20).

- [25] Ryan Koh, Adrian Nachman, and José Zariffa. «Use of spatiotemporal templates for pathway discrimination in peripheral nerve recordings: A simulation study». In: *Journal of neural engineering* 14 (Dec. 2016), p. 016013. DOI: 10.1088/1741-2552/14/1/016013 (cit. on p. 20).
- [26] Ryan Koh, Adrian Nachman, and José Zariffa. «Classification of naturally evoked compound action potentials in peripheral nerve spatiotemporal recordings». In: *Scientific Reports* 9 (Dec. 2019). DOI: 10.1038/s41598-019-47450-8 (cit. on p. 20).
- [27] Yuang Tang, Brian Wodlinger, and Dominique Durand. «Bayesian Spatial Filters for Source Signal Extraction: A Study in the Peripheral Nerve». In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 22 (Mar. 2014), pp. 302–311. DOI: 10.1109/TNSRE.2014.2303472 (cit. on p. 20).
- [28] Brian Wodlinger and Durand Durand. «Localization and Recovery of Peripheral Neural Sources With Beamforming Algorithms». In: *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* (Nov. 2009), pp. 461–468 (cit. on p. 20).
- [29] W Tesfayesus and Durand Durand. «Blind source separation of peripheral nerve recordings». In: *Journal of neural engineering* 4 (Oct. 2007), S157–67. DOI: 10.1088/1741-2560/4/3/S03 (cit. on p. 20).
- [30] Jordi Badia, Arán Pascual-Font, Meritxell Vivó, Esther Udina, and Xavier Navarro. «Topographical distribution of motor fascicles in the sciatic-tibial nerve of the rat». In: *Muscle and Nerve* 42.2 (2010), pp. 192–201. ISSN: 10974598. DOI: 10.1002/mus.21652 (cit. on p. 22).
- [31] H. Schmalbruch. «Fiber composition of the rat sciatic nerve». In: *The Anatomical Record* 215.1 (1986), pp. 71–81. ISSN: 10970185. DOI: 10.1002/ar.1092150111 (cit. on p. 22).
- [32] A. L. Hodgkin and A. F. Huxley. «A quantitative description of membrane current and its application to conduction and excitation in nerve». In: *The Journal of Physiology* 117.4 (1952), pp. 500–544. ISSN: 14697793. DOI: 10.1113/jphysiol.1952.sp004764 (cit. on pp. 23, 26).
- [33] C. C. McIntyre, A. G Richardson, and W. M. Grill. «Modelling the effects of electric fields on nerve fibres: Influence of the myelin sheath». In: *Medical and Biological Engineering and Computing* 38.4 (2000), pp. 438–446. ISSN: 01400118. DOI: 10.1007/BF02345014 (cit. on pp. 23, 26, 32, 33, 87).
- [34] J Hope, F Vanholsbeeck, and A. McDaid. «A model of electrical impedance tomography on peripheral nerves for a neural-prosthetic control interface». In: (2017). arXiv: 1711.04377. URL: <https://arxiv.org/ftp/arxiv/papers/1711/1711.04377.pdf> <http://arxiv.org/abs/1711.04377> (cit. on pp. 23, 24).

- [35] A. Weerasuriya, R. A. Spangler, S. I. Rapoport, and R. E. Taylor. «AC impedance of the perineurium of the frog sciatic nerve». In: *Biophysical Journal* 46.2 (1984), pp. 167–174. ISSN: 00063495. DOI: 10.1016/S0006-3495(84)84009-6 (cit. on p. 23).
- [36] James B. Ranck and Spencer L. BeMent. «The specific impedance of the dorsal columns of cat: An anisotropic medium». In: *Experimental Neurology* 11.4 (1965), pp. 451–463. ISSN: 10902430. DOI: 10.1016/0014-4886(65)90059-2 (cit. on p. 24).
- [37] Kirill Aristovich, Matteo Donegá, Camille Blochet, James Avery, Sana Hannan, Daniel J. Chew, and David Holder. «Imaging fast neural traffic at fascicular level with electrical impedance tomography: Proof of principle in rat sciatic nerve». In: *Journal of Neural Engineering* 15.5 (2018). ISSN: 17412552. DOI: 10.1088/1741-2552/aad78e (cit. on pp. 24, 27, 28).
- [38] M. L. Hines and N. T. Carnevale. «The NEURON Simulation Environment». In: *Neural Computation* 9.6 (1997), pp. 1179–1209. ISSN: 08997667. DOI: 10.1162/neco.1997.9.6.1179 (cit. on pp. 25, 31).
- [39] Ilya Tarotin, Kirill Aristovich, and David Holder. «Model of impedance changes in unmyelinated nerve fibers». In: *IEEE Transactions on Biomedical Engineering* 66.2 (2019), pp. 471–484. ISSN: 15582531. DOI: 10.1109/TBME.2018.2849220 (cit. on pp. 25–28).
- [40] J. D. Sweeney, J. T. Mortimer, and D. Durand. «Modeling of Mammalian Myelinated Nerve for Functional Neuromuscular Stimulation.» In: *IEEE/Engineering in Medicine and Biology Society Annual Conference* January 1987 (1987), pp. 1577–1578 (cit. on p. 25).
- [41] *Model DB*. URL: <https://senselab.med.yale.edu/modeldb/> (cit. on pp. 26, 32).
- [42] B. Frankenhaeuser and A. F. Huxley. «The action potential in the myelinated nerve fibre of *Xenopus laevis* as computed on the basis of voltage clamp data». In: *The Journal of Physiology* 171.2 (1964), pp. 302–315. ISSN: 14697793. DOI: 10.1113/jphysiol.1964.sp007378 (cit. on p. 26).
- [43] Warren M. Grill. «Modeling the effects of electric fields on nerve fibers: Influence of tissue electrical properties». In: *IEEE Transactions on Biomedical Engineering* 46.8 (1999), pp. 918–928. ISSN: 00189294. DOI: 10.1109/10.775401 (cit. on p. 26).

- [44] Pierre Sacre, Sridevi V. Sarma, Yun Guan, and William S. Anderson. «Electrical neurostimulation for chronic pain: On selective relay of sensory neural activities in myelinated nerve fibers». In: *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS 2015-Novem* (2015), pp. 4705–4708. ISSN: 1557170X. DOI: 10.1109/EMBC.2015.7319444. arXiv: 1507.02716 (cit. on p. 26).
- [45] Harilal Parasuram, Bipin Nair, Egidio D’Angelo, Michael Hines, Giovanni Naldi, and Shyam Diwakar. «Computational modeling of single neuron extracellular electric potentials and network local field potentials using LFPsim». In: *Frontiers in Computational Neuroscience* 10.Jun (2016), pp. 1–13. ISSN: 16625188. DOI: 10.3389/fncom.2016.00065 (cit. on p. 26).
- [46] T. Hayami, K. Iramina, and X. Chen. «Computer simulation of nerve conduction study of a sural nerve to evaluate human peripheral nervous system». In: *IFMBE Proceedings* 63 (2018), pp. 461–465. ISSN: 16800737. DOI: 10.1007/978-981-10-4361-1\_78 (cit. on p. 27).
- [47] Adrien Rapeaux, Konstantin Nikolic, Ian Williams, Amir Eftekhari, and Timothy G. Constandinou. «Fiber size-selective stimulation using action potential filtering for a peripheral nerve interface: A simulation study». In: *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS 2015-Novem* (2015), pp. 3411–3414. ISSN: 1557170X. DOI: 10.1109/EMBC.2015.7319125 (cit. on p. 27).
- [48] Nicole A. Pelot, Christina E. Behrend, and Warren M. Grill. «On the parameters used in finite element modeling of compound peripheral nerves». In: *Journal of Neural Engineering* 16.1 (2019). ISSN: 17412552. DOI: 10.1088/1741-2552/aaeb0c (cit. on p. 27).
- [49] Ilya Tarotin, Kirill Aristovich, and David Holder. «Simulation of impedance changes with a FEM model of a myelinated nerve fibre». In: *Journal of Neural Engineering* (2019). ISSN: 17412552. DOI: 10.1088/1741-2552/ab2d1c. arXiv: 1903.04948 (cit. on pp. 27, 28).
- [50] Carl H. Lubba, Yann Le Guen, Sarah Jarvis, Nick S. Jones, Simon C. Cork, Amir Eftekhari, and Simon R. Schultz. «PyPNS: Multiscale Simulation of a Peripheral Nerve in Python». In: *Neuroinformatics* 17.4 (2019), p. 629. ISSN: 15590089. DOI: 10.1007/s12021-019-09421-2 (cit. on pp. 28, 32).
- [51] *Comsol Documentation*. URL: <https://www.comsol.com/documentation> (cit. on p. 37).



- [52] Li Jiang, Xinyuan Zhao, Yue Fei, Dongdong Yu, Jun Qian, Jinguang Tong, Guangdi Chen, and Sailing He. «The effects of magnetic fields exposure on relative permittivity of saline solutions measured by a high resolution SPR system». In: *Scientific Reports* 6 (Apr. 2016), p. 25111. DOI: 10.1038/srep25111 (cit. on p. 47).
- [53] *Tissue Frequency Chart by IT'IS Foundation*. URL: <https://itis.swiss/virtual-population/tissue-properties/database/tissue-frequency-chart/> (cit. on pp. 48, 49).
- [54] *Intro to Analyzing Electromagnetic Fields with COMSOL Multiphysics<sup>®</sup>*. URL: <https://www.comsol.com/video/intro-to-analyzing-electromagnetic-fields-with-comsol-multiphysics> (cit. on p. 50).
- [55] *Sim4Life - Zurich MedTech*. URL: <https://zmt.swiss/sim4life/> (cit. on p. 64).