TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

D I P L O M A R B E I T

Computer Simulation of Retinal Ganglion Cell Stimulation Comparing Varying Electrode Spacings

Ausgeführt am

INSTITUT FÜR ANALYSIS UND SCIENTIFIC COMPUTING

der Technischen Universität Wien

unter der Anleitung von

Ao.Univ.Prof. Dipl.-Ing. DDDr. Frank Rattay

und

und Dipl.-Ing. Dr. techn. Paul Werginz

durch

Anton Watzinger

Wien, 30.05.2017

Prof. DDDr. Frank Rattay	Anton Watzinger

Acknowledgements

First I want to thank Ao.Univ.Prof. Dipl.-Ing. Dr.rer.nat. Dr.techn. Dr.scient.med. Frank Rattay, whose support and expertise have guided me throughout my studies. The door to Prof. Rattay's office was always open whenever I had a question about my research or ran into a trouble spot and so made this thesis possible. His devotion to his students has been greatly appreciated.

Furthermore, I would like to thank Dipl.-Ing. Dr.techn. Paul Werginz for his supervision, insightful comments and hard questions.

Also, I want to thank my friends and companions throughout my studies; Dipl.-Ing. Lukas Sölkner and Mag. Petra Chaberova, who kept me company during countless hours of work.

Finally, I must express my very profound gratitude to Mag. Silvia Mezin, who always supported and inspired me, showed great understanding, and never complained while I wrote this thesis.

Kurzfassung

Künstlich erzeugte elektrische Signale in Retina Implantaten unterscheiden sich signifikant von den Signalen, die auf natürlichem Wege zustande kommen, was dazu führt, dass sie schwer vom Gehirn interpretiert werden können. Generell können Computer Modelle, insbesonders das für diese Arbeit entwickelte, das Verhalten von neuronalen Netzwerken simulieren. Zum Beispiel um die Auswirkungen von verschiedenen Größen von Elektrodenkonfigurationen zu vergleichen. Allerdings sind bessere Simulationsmethoden erforderlich, um diese Mechanismen besser zu verstehen und die medizinischen Anwendungen zu verbessern. Ziel dieser Arbeit ist es, den optimalen Elektrodenabstand eines Retina Implantats zu ermitteln. Damit können Implantate an Patienten, die an degenerativen Retinaerkrankungen leiden, sicher eingesetzt werden.

Der Energieverbrauch einer Säugetier-, Amphibien- und Kopffüßer- (Tintenfisch) Neuronalmembranen wurde verglichen. Die Membran des Säugetieres wurde nach dem Fohlmeister et al 2010 Modell modelliert, die Membran der Amphibien nach dem Fohlmeister et al 1997 Modell und die Membran des Tintenfisches nach dem Hodgkin und Huxley Modell. Des weiteren wurde der Energieverbrauch der Membranen analysiert, einschließlich gleichzeitiger stimulierenden und hemmenden Eigenschaften. Für diese Arbeit wurde ein Computermodell mit MATLAB und der Finite-Elemente-Software Comsol Multiphysics implementiert, um die Aktivierung von retinalen Ganglionzellen während der epi-retinalen Stimulation vorauszuberechnen. Die neuronale Aktivierung, mathematisch dargestellt durch das Membranmodell von Fohlmeister et al 2010 wurde an verschiedenen Teilen des Neurons (elektrisch sensible Natrium Band und proximal gelegene Axon) mit verschiedenen Elektrodenabständen simuliert.

Der Vergleich zwischen den Membranen verschiedener taxonomischer Klassen zeigt einen teilweise erhöhten Energieverbrauch in Natrium- und Kaliumkanälen der weiterentwickelten Säugetiermembran im Vergleich zur

Amphibienmembran. Dieser wird aber durch eine verbesserte Synchronisation zwischen den verschiedenen Ionenkanaltypen kompensiert und ergibt eine bessere Gesamteffizienz der Säugetiermembran. Die Simulation verschiedener Elektrodenabstände zeigte einen Zusammenhang zwischen dem Elektrodenabstand und der benötigten Stimulationsenergie. Weiters wurde ein idealer Elektrodenabstand gefunden, bei dem niedrige Energien benötigt werden, und dabei kein wertvoller Platz auf dem Implantat verschwendet wird.

Die gewählten Membranmodelle und das Aktivierungsmodell zusammen mit den oben erklärten Methoden und den durchgeführten Simulationen erwiesen sich als passend, um das Ziel dieser Arbeit zu erreichen. Aus dem Wissen über die Beschaffenheit des Nervensystems und aus den Fortschritt der heutigen Wissenschaft profitieren die Patienten, die an retinalen Degenerationskrankheiten leiden. Sie bekommen die Chance auf eine teilweise Wiederherstellung ihres Sehvermögen.

Abstract

The artificially generated electrical signal, in retinal prostheses is very different from the natural neuronal signal and difficult for the brain to interpret. Computer models in general -the present one in particular- have the ability to simulate the behaviour of the neuronal network in ways such as comparing different sizes of stimulating electrode configuration. However, better simulation methods are needed to fully understand these neuronal mechanisms and improve their medical applications. The objective of this thesis is to identify the optimal electrode spacing of a retinal implant that can be safely used, so that patients suffering from retinal degenerative diseases can profit from better-constructed implants.

For this purpose, the energy consumption of mammalian, amphibian and cephalopodal (squid) neuronal membranes have been compared. The mammalian membrane was modelled after the Fohlmeister et al 2010 model, the amphibian membrane after Fohlmeister et al 1997 model and the squid membrane after Hodgkin and Huxley. Furthermore, their detailed energy usage was analysed, including simultaneous stimulating and inhibiting properties. Especially for this thesis, a computer model has been implemented using MATLAB and the finite element software Comsol Multiphysics in order to predict the activation of retinal ganglion cells during epi-retinal stimulation. The activation, driven by the membrane model of Fohlmeister et al 2010 was simulated on different parts of the neuron (sodium channel band and distant axon) with various electrode spacings.

The comparison between the membranes of different taxonomic classes showed a partially increased energy consumption in sodium and potassium channels of the higher developed mammalian membrane compared to the amphibian membrane, which was compensated by improved synchronization between the different ion channel types, allowing for better cross efficiency. The simulation of various electrode spacings showed a dependency between the stimulating energy on the neuron and the electrode spacing. Further-

more, an ideal electrode spacing was identified, where the neuron was able to stimulate with least amount of energy, without wasting valuable space on the electrode carrier.

The chosen membrane models and activating model, together with the methods and the performed simulations, proved to be the right combination needed in order to reach the objective of this thesis.

Benefiting from the knowledge of the characteristics of the nervous system and from the capacity of today's science, patients suffering from retinal degeneration diseases are given the chance to have their sense of vision restored.

Contents

1 Introduction **1**

1.1 Motivation 1

1.2 Implementation 2

2 Background **3**

2.1 Human Eye 3

2.2 Neuronal Cells 6

2.2.1 Soma 7

2.2.2 Dendrite 8

2.2.3 Axon 8

2.2.4 Retinal Neurons 9

2.3 Action Potential 11

2.3.1 Membrane Potential and Gating 11

2.3.2 Action Potential 12

2.3.3 Refractory Period 12

2.3.4 Graded Potential 13

2.4 Retinal Prosthesis 13

2.4.1 Sub-retinal Stimulation 14

2.4.2 Epi-retinal Stimulation 15

2.4.3 Supra-choroidal Stimulation 15

2.5 Subtypes of Computational Retina Models 16

2.5.1 Single-Compartment Models 16

2.5.2 Morphologically-Realistic Models 16

2.5.3 Block-Compartment Models 16

2.5.4 Continuum Models 17

2.5.5 Block-Structured Models 17

2.5.6 Discrete-Neuronal Network Models 17

2.6 Basics of Electrical Stimulation 18

2.6.1 Electrical Network Model 18

2.6.2 Activating Function 19
2.7 Membrane Models 20
2.7.1 Hodgkin Huxley Model (1952) 20
2.7.2 Fohlmeister and Miller Model (1997) 23
2.7.3 Fohlmeister, Cohen and Newman Model (2010) 28

3 Comparison of Membrane Properties 31
3.1 Ionic Current Flow 31
3.2 Double-Pulse Experiment 36

4 Electrode Configuration Study 41
4.1 Methods 41
4.1.1 Retinal Tissue Model 42
4.1.2 Retinal Ganglion Cell 43
4.1.3 Electrode Model 45
4.1.4 Stimulation Pulse 48
4.1.5 Finite Element Method (FEM) 49
4.2 Electrode Distance Optimization 55
4.2.1 Potential Shape on Various Spacings 56
4.2.2 Axon Stimulation 60
4.2.3 Sodium Channel Band Stimulation 65
4.3 Selective Stimulation 70
4.3.1 Activation Map 71

5 Discussion 76
5.1 Membrane Potential 76
5.2 Electrode Configuration Study 77

Bibliography 78

Appendix A Resources 84
A.1 Sourcecode 84
A.2 Ca-Kinetics Correction factor 103

Appendix B FEM Mesh 105
B.1 FEM Mesh of Bulk Volume 105

Abbreviations and Symbols

A	stochastic gating variables for A-type K channels (activating)
A_h	stochastic gating variables for A-type K channels (inactivating)
AH	axon hillock
AP	action potential
ATP	adenosine triphosphate
BC	boundary conditions
c	stochastic gating variables for calcium channels
C_m	membrane capacitance
DA	distal axon
F	Faraday constant
FEA	finite element analysis
FEM	finite element method
$g_K, g_{K,A}, g_{K,Ca}, g_L, g_{Na}$	membrane conductance of $I_K, I_{K,A}, I_{K,Ca}, I_L, I_{Na}$
h	stochastic gating variables for Na channels (inactivating)
m	stochastic gating variables for Na channels (activating)
n	stochastic gating variables for K channels
i	current density
I	current
I_C	capacitive current
I_{Ca}	calcium current
I_{ion}	ion current passing through the membrane
I_K	potassium current
$I_{K,A}$	A-Type potassium current
$I_{K,Ca}$	calcium activated potassium current
I_L	leakage current
I_{Na}	sodium current

I_r	ohmic current
I_{stim}	stimulus current
R	universal gas constant
RC model	resistor - capacitor model
RGC	retinal ganglion cell
SOCB	sodium channel band
T	absolute temperature
V	membrane potential, relative to V_{rest}
V_{Ca}	equilibrium potential of Ca concentration
V_e	extracellular potential
V_i	intracellular potential
V_K	equilibrium potential of K concentration
V_l	leakage potential
V_{Na}	equilibrium potential of Na concentration
V_{rest}	resting potential
$[Ca^{++}]_e$	extracellular Ca concentration
$[Ca^{++}]_{res}$	resting cystosolic Ca concentration
λ_r	calcium buffer surface to volume ratio
τ_{Ca}	calcium removal time constant

Chapter 1

Introduction

1.1 Motivation

About 1.5 million people worldwide suffer from degenerative disease of photoreceptor cells [Palanker et al., 2005]. While the outer photoreceptor cells degenerate from diseases like age-related macular degeneration (AMD) and retinitis pigmentosa (RP), a significant amount of the inner Retinal Ganglion Cells (RGC) and the optical nerve might remain intact. We know that the retina is a complex network containing a vast amount of cell types with a clear structure. This structure allows us to identify functional vectors which can be simulated in computer models, and apply these results in electrically stimulated clinical applications. This innovation raises the possibility to generate meaningful vision for blind patients, significantly improving the lives of affected individuals. The overall goal is to enable visual perception in the blind, perception which is comparable to our natural sense of vision. While visual prostheses have been available in the field for over a decade, the above-mentioned goal is still far from reach [Yue et al., 2015].

A key factor for meaningful stimulation is a high amount of parallel stimulation channels on the comparably small region of retinal tissue [Zrenner et al., 2011]. Based on well-known histological properties [Karwoski et al., 1985] and experience of neuronal stimulation in other fields [Merrill et al., 2005], the simulation concept developed by Greenberg and co-workers led to a better understanding of the entire retinal network [Greenberg et al., 1999].

Aside from more detailed anatomical data cell membrane models for mammals also became available [Fohlmeister et al., 2010]. Relatively new findings regarding a sodium channel band (SOCB) which is sensitive to extracellular electric stimulation [Fried et al., 2009, Jeng et al., 2011], com-

bined with better understanding of electrodes in retinal prosthesis [Wilke et al., 2011], allow the development of more precise computational models, resulting in more realistic simulations.

This thesis analyses different electrode distance configurations in the context of space demand, energy consumption and selectivity at tissue-level. This is examined in two circumstances: the stimulation of the plane axon [Resatz, 2005] and the more sensitive sodium channel band [Fried et al., 2009]. Furthermore the energy consumption was evaluated on cell-level.

1.2 Implementation

In order to achieve a dynamic way of simulating different parameters, a simulation software framework was developed, using MATLAB 8.5.0.197613 (R2015a)[1] and Comsol Multyphysics 4.4.0.150[2]. Plots were generated by matlab2tikz 0.6.0[3] and compiled by LuaTeX 0.80[4]. All simulations were optimised for consumer computers working on Linux with Intel core-i7 architecture and 8 GB memory. If not explicitly mentioned otherwise, each calculation was completed within a maximum of a few minutes.

Moreover, the above-mentioned program tries to provide ways for further neuronal simulations on different fields. To allow other people to work with or modify this new software, all sources are available online (Chapter A). Additionally, a very common approach from computer science, 'plug-ins' has been applied. Only very basic concepts of neuronal stimulation, e.g. the activating function and the electrical exchange between compartments, have been implemented in the core of the system. Other parts which are more specific to the simulation, like the ion channel densities, the membrane model or the stimulation electrode position, were added by the use of plug-ins.

The newly developed software with these precise plug-ins forms the environment necessary to see separate variations either by using different membrane models (like Hodgkin and Huxley compared to Fohlmeister and Miller [Fohlmeister and Miller, 1997a, Hodgkin and Huxley, 1952]) on identical nerve structures or the use of the same membrane model on different neuronal morphologies.

[1] http://de.mathworks.com/products/matlab/
[2] https://www.comsol.com/
[3] http://www.mathworks.com/matlabcentral/fileexchange/22022-matlab2tikz
[4] http://www.luatex.org/

Chapter 2

Background

2.1 Human Eye

To provide the necessary physiological background for this thesis, the following anatomy of the human eye description reviews the 'Atlas der Anatomie' [Kahle, 1991] and "The Organization of the Retina and Visual System" [Kolb et al., 1995]. The focus is set on neuronal and visual aspects, but histological and muscular properties have been left out.

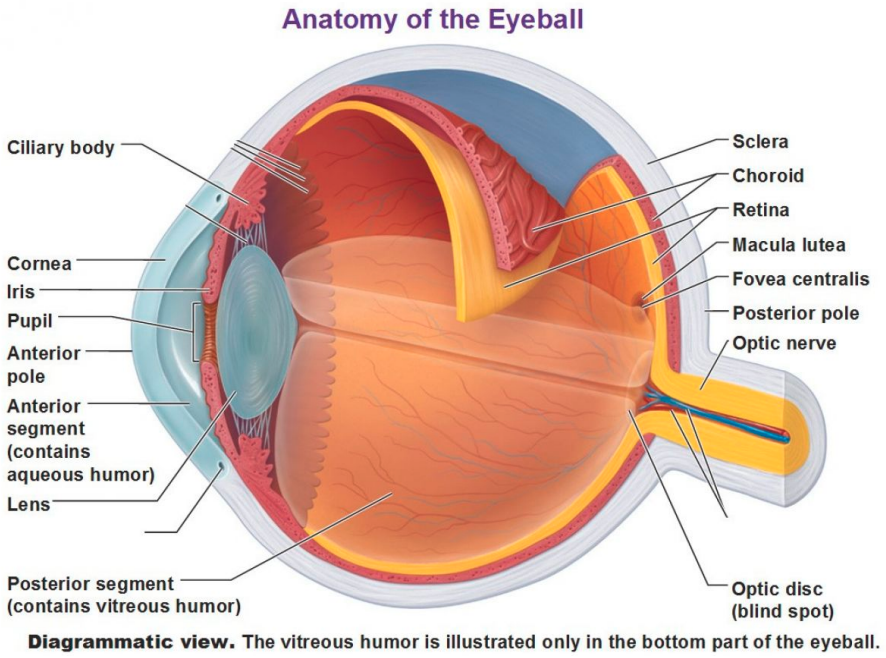


Figure 2.1: Cross section of a human eyeball [Kolb et al., 1995]

Optical System

On the front face of the eyeball, as light's entry point, the transparent cornea is placed. It has the optical effect of a convex collecting lens. The cornea covers the anterior chamber (or anterior segment), which contains the aqueous humour and covers both the iris and the pupil. The cornea has an optical power of over 40 dioptre and is the highest focusing factor in the human optical pathway.

Directly behind the anterior chamber is located the iris, which serves as aperture. The iris consists of 2 layers, where the first layer (stroma) contains blood vessels and nerves, and the second layer holds the pigmented cells. These cells are responsible for the appearance of colour, ranging from blue (low pigment level) to brown (high pigment level). The centrally placed opening of the iris -the pupil- controls the amount of light entering into the eye by changing the size.

The light passing through the pupil is focused by the crystalline lens. The curvature of the lens and therefore the focus on objects of various distance, is adapted by the ciliary muscle.

The next stop on the light's journey is the inner volume of the eye, the vitreous body. It is filled with an aqueous gel, the vitreous humour.

Between the vitreous body and the choroid, the retina is placed. The retina adjoins the choroid, a thin skin which contains many blood vessels and provides the majority of blood flow to the eye

The outermost layer of the eye is the sclera, a thick collagen skin which maintains the spherical form of the eye and is responsible for the white eye.

Neuronal System

Continuing from the vitreous body at the back of the eye, the light passes through the transparent (and between $80\mu m$ and $320\mu m$ thick) retina until it is perceived by photoreceptor cells [Kolb et al., 1995]. This arrangement, where the light passes through the retina before getting detected by the photoreceptor cells, is called "inverted retina". It can be found in all vertebrae retinas, while verted retinas can be found in invertebrates such as squids. The process of passing through the retina layers however, does not significantly disturb the light.

The neuronal cell is formed by the soma (cell body), axon and dendrite and will be described in detail in Chapter 2.2. The innermost cells in the retina are the multipolar RGC in the ganglion cell layer. They have rather

short dendrites, reaching out to the plexiform layer, but their long axons form the optical nerve. These RGC are the main target cells in this study.

The dendrites of the RGC are connected in the inner plexiform layer to the axons of the bipolar cells. The bipolar cells' somas are located in the inner nuclear layer and aim with their dendrites towards the photoreceptor cells. Within the inner nuclear layer, the somas of the horizontally connected amacrine cells are located. Their dendrites reach out to the photoreceptor cells and connect to neurons in the inner plexiform layer. On the other hand, the dendrites of the bipolar cells are connecting to the photoreceptor cells.

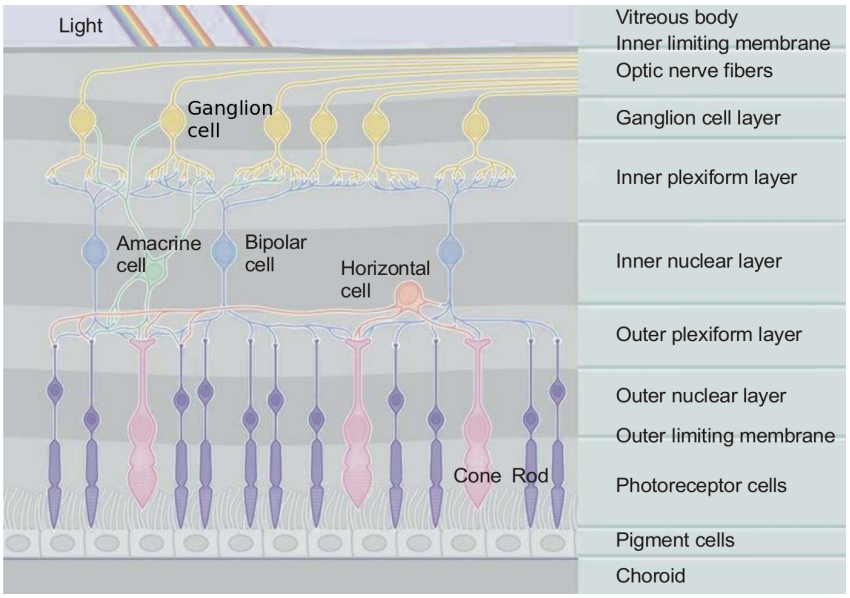


Figure 2.2: Schematic connection of retinal neurons and retinal layers. Modified from [Resatz, 2005]

Connected to the choroid, the outermost layer is the retinal pigment epitelium, which has important functions in maintaining the metabolism of photoreceptor cells.

The retina is far more than a simple sensor. As part of the central nervous system, its complex connections, multiple neuron types and total of 10 layers all process the images perceived by the eye. Also noticed in the retina, is some functional clustering, where cells can inhibit the activation of neighbour cells. This influence on neighbour cells is important for perception of contrasts, for instance.

The retina has 2 regions worth mentioning:

1. **Fovea centralis**
   The fovea centralis is the region in the retina with the best vision. The pigment epithelium and the photoreceptor layers are thickened in this region, hence the plexiform layers are thinned. This leads to less attenuation and further, to the best access for light to the photoreceptor cells. Additionally, the photoreceptor layer contains only cones (for colour vision) in this region, and as a result contains the highest concentration of cones in the entire retina. Contrary to other regions, each cone is connected to only one single bipolar cell. Only mammals have this region, as amphibian retinas lack a fovea centralis.

2. **Optic disc**
   The optical nerve, which is formed on the inner surface of the eye from RGC axons, needs to exit the eyeball through the retina. This spot at which it exits, has no photoreceptor cells, therefore no light can be perceived. This is why it is commonly referred to as the "blind spot". The optical nerve transmits the perceived information to the visual cortex in the back side of the brain.

It should be mentioned, that, unlike most other neuronal networks, this is a 'one-directional' system, which means that there is no feedback from the brain.

2.2 Neuronal Cells

The cell physiology is reviewed from "Anatomy and Physiology" [Betts, 2013] and "Electrical Nerve Stimulation" [Rattay, 1990].

Neurons are the most important cells of the nervous system, because they are responsible for generating, propagating and processing electrical signals. They consist in general of the soma, axon and dendrite, but appear in many other different morphologies. Still, they can be classified according to their structure, function or neurotransmitter. An information signal (in the form of action potentials -AP-), flows only in one direction. It arrives in the neuron through the dendrites, passes through the soma, further down to

the axons, and exits at the synapses. A neuron's polarity, describe how many processes it has. Unipolar neurons, like the photoreceptor cells, have only an axon, while bipolar neuron possess an axon and a dendrite. Multipolar neurons have one axon and two or more dendrites. Since the RGCs have multiple dendrites which connect to bipolar cells and amacrine cells, RGC are multipolar cells.

Besides the neurons which simply transmit signals to other neurons (interneurons), there are two types of specialized neurons in the central nervous system (CNS). First, there are the sensory neurons, which connect to receptors cells. These receptor cells can be stimulated by chemical or physical energy such as pressure, heat or light. Taste receptors, inner hair cells and photoreceptor cells can all be found in this category. Second, there are the motoneurons (or motor neurons), which control muscle movement.

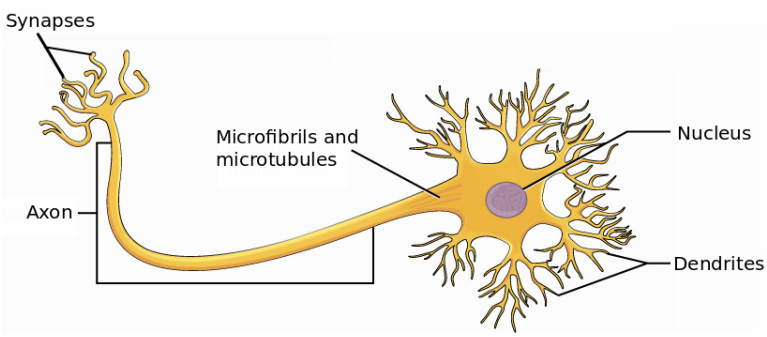


Figure 2.3: Neuron with dendrites, soma and non-mylenated axon. Adapted from [Betts, 2013].

Neurons are not the only cell type in neuronal tissue. There is a second group, the 'glial cells' (or non-neuronal cells). Cells of this type do not carry electrical signals, but support the neurons, e.g. the Oligodendrocyte and Schwann cells build the myeline sheat. Latest research shows that the human brain consists of about $8.6 \cdot 10^{10}$ neurons and $8.4 \cdot 10^{10}$ non-neurons (glial cells) [Azevedo et al., 2009].

2.2.1 Soma

Each neuron has a soma (also known as 'cell body'). This soma contains most of the organelles including the nucleus. Inside the membrane is a liquid called cytoplasm, as well as a cytoskeleton which maintains stability (among other functions). The soma extends the processes, which are either dendrites

or axons. All these parts are then covered by the cell membrane. The size of the soma varies in different neuronal cells.

2.2.2 Dendrite

Dendrites (rarely dentrite) act as the input region for signals of other cells, and form a dendritic tree with up to 5.000 branches. Dendrites possess a wide network of cytoplasmic processes which allow them to receive signals from a wide range of different cells. This high amount of connections is possible because of the 3D structure of the tree. The RGC, for instance, receive through their dendrites the signals of the bipolar cells and amacrine cells. Like axons, dendrites are processes that originate from in the cell membrane of the soma.

2.2.3 Axon

The axon is a nerve fibre which emerges from the cell body. It is filled with axoplasm, the cytoplasm of axons. Axons start with the axon hillock (also known as initial segment), which tapers from soma to distal axon. The main purpose of axons is to carry electrical potential from soma to synapses. In the case of vertebrates, both myelinated and non-myelinated axons can be found. In contrast to non-myelinated cells, myelinated cells have a faster propagation speed, even on small axon diameters. Since the RGC axons are not covered by a myelin sheath, this subsection focuses on non-myelinated axons.

The length of an axon varies between a few micrometers and several meters, among different neuron types. Some axons, especially RGC axons, have a region with a SOCB. In this region there is a highly increased number of sodium channels, which significantly increases the sensitivity to electrical stimulation. While this finding is relatively new, the fact that axons are the most excitable part of the neuron has been known since the late 1970s [Nowak and Bullier, 1998]. There are however, other neuronal parts that are also excitable by extracellular stimulation. Due to the SOCB being part of the axon, the part furthest from the SOCB is in this thesis referred to as distal axon (DA). Axons are responsible for delivering electrical signals to distant neurons or muscles.

Synapses

At the ending of a single axon, up to 200.000 synapses are found. These synapses are the connections to other cells, and can be based on chemical signals, so-called neurotransmitters, or electrical signals. Synapses for controlling muscles, the neuromuscular synapses, have only excitatory potential, while other synapses can have either inhibitory or excitatory interfaces. The junction between the synapse and the targeting cell is formed as a tiny gap, namely the synaptic cleft, which is usually about $20nm$ in electric synapses and $100nm$ in chemical synapses [Betts, 2013, Pfützner, 2012].

2.2.4 Retinal Neurons

The retina contains a vast amount of different neuron types, but in general they can be divided into the following five classes, as depicted in Figure 2.2: photoreceptor cells (rod and cones), amacrine cells, bipolar cells, horizontal cells and RGC [Seung and Sümbül, 2014].

Photoreceptor Cells

Photoreceptor cells are aligned as an array and can be categorised into rods (for brightness) and cones (for colour). Cones, which are responsible for vision in well-lit areas (photopic vision) allow humans to see light over a wavelength spectrum from $380nm$ to $720nm$. The vision of brightness is often referred as scotopic vision [Betts, 2013, Kolb et al., 1995].

Rods The light-sensitive side of rods contains a stack of membrane-bound discs. These discs contain the light-sensitive protein rhodopsin. The incoming light passing through this stack allows the rod an extremely high light sensitivity. A single photon can trigger an action potential in a rod's corresponding RGC. Rhodopsin has its highest sensitivity to a wavelength around $500nm$, and therefore the low-light vision is in greyscale. The human retina has around 120 million rods [Betts, 2013].

Cones In humans, three subgroups of conical shaped cones called opsins, and ten types of cones are known. Each type is sensitive to light of a particular wavelength, which allows the brain to identify concentrations of different wavelengths and therefore perceive colours. The body of a cone is slightly longer than the rod's cell body, reaching out to the pigment epithelium. There are around 6 million of these cells in the human eye [Betts, 2013].

Amacrine Cells

The most diverse class of cells in the retina are the amacrine cells, with 29 known subtypes. These cells interconnect with the photoreceptor cells, bipolar cells and ganglion cells, and are essential for early image processing before the AP is produced by the RGC. The structure and function of these cells are manifold: some types of amacrine cells are horizontally directed, while some are vertically directed; some are part of ON, while others are part of OFF pathways. This derives if the cell is activated by light onset (ON-cell) or by light offset (OFF-cell). [Betts, 2013, Kolb et al., 1995].

Horizontal Cells

In the mammalian retina, three different types of horizontal cells are known, where two types connect exclusively to cones and one, to rods. They play an important role in signal processing: Horizontal cells average and feedback photoreceptor signals [Resatz, 2005].

Bipolar Cells

Bipolar cells connect on their dendrite side to photoreceptor cells and on their axon side, to ganglion cells. While a special type of bipolar cell-the 'rod bipolar cell'-connects only to rods, 10 to 12 different bipolar cell types were identified, which connect only to cones. On average, 15 to 45 rods share one rod bipolar cell. Bipolar cells also have ON and OFF types, and have a high influence on the greater image processing activity of RGC [Baden et al., 2013].

Retinal Ganglion Cells

RGC are the innermost retina cells, and are responsible for generating the output signal of the retina by evaluating the signals from bipolar cells, as well as the inhibitory synapses from amacrine cells. The axons of RGCs merge at the optic disc and exit the eye as the optic nerve. In mammals, between 13 and 17 different types have been discovered [Fohlmeister et al., 2010]. Like in amacrine and bipolar cells, RGC can be found in both the ON and OFF pathways [Betts, 2013, Kolb et al., 1995].

2.3 Action Potential

2.3.1 Membrane Potential and Gating

The action potential originates in the cell membrane. When unexcited, the cell membrane stays in a resting potential which is not 0, but around $-70mV$ relative to the extracellular space. Varying concentrations of ions in the cytosol (inner neuron) and in the extracellular fluid are responsible for this potential difference. These concentration gradients are maintained by ion channels in the cell membrane. The cell membrane is a lipid double-layer, with active and passive pores. The active pores ATP (adenosine triphosphate) serve as an energy source, pumping sodium and potassium ions through the membrane.

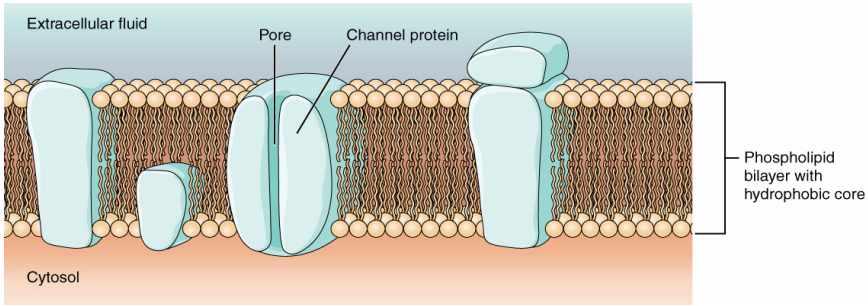


Figure 2.4: Scheme of a cell membrane [Betts, 2013]

Ion channels can be distinguished between mechanical, chemical, voltage and random gated channels. There are voltage gated sodium channels and mechanically gated calcium channels. A-type potassium channels are chemically or ligand gated, and the leakage is randomly gated. Mechanically gated ion channels can be found in sensory neurons. The Goldmann equation (Equation 2.1) can be used to estimate the resting potential according to the intracellular and extracellular ion concentrations [Fohlmeister and Miller, 1997a, Goldman, 1943, Pfützner, 2012].

$$E_m = \frac{RT}{F} \cdot ln \frac{g_K \cdot [K^+]_e + g_{Na} \cdot [Na^+]_e + g_{Cl} \cdot [Cl^-]_i}{g_K \cdot [K^+]_i + g_{Na} \cdot [Na^+]_i + g_{Cl} \cdot [Cl^-]_e} \tag{2.1}$$

$$-70mV = 58mV \cdot ln \frac{40 \cdot 5.5 + 1 \cdot 150 + 0 \cdot 9}{40 \cdot 150 + 1 \cdot 15 + 0 \cdot 125} \tag{2.2}$$

As shown in Equation 2.2, during resting state, the concentration of K^+ ions is higher inside the cell than outside, while the concentration of Na^+ ions is higher outside the cell than inside.

2.3.2 Action Potential

A neuronal cell can be excited if the stimulus overcomes a specific threshold (mostly around $-50mV$). If this threshold is not reached, the potential will fall back to the resting potential, without initiation and consequent propagation of an AP. This is the so-called all-or-none or all-or-nothing law.

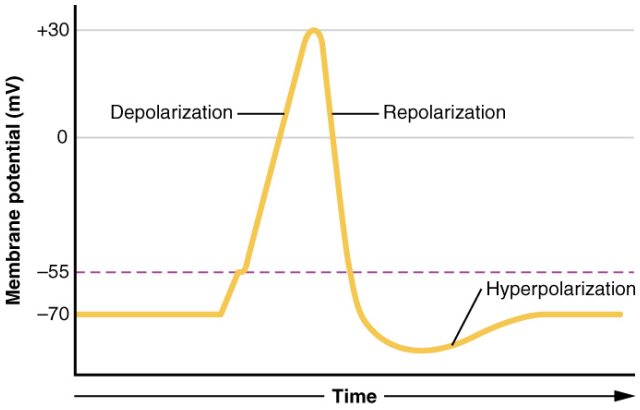


Figure 2.5: Time response of an action potential. The dotted line denotes the approximate threshold [Betts, 2013].

If this threshold is exceeded, the sodium channels open, which causes an inflow of Na^+ ions across the membrane. Because of the positive charge of Na^+ ions, the voltage in the cell rises, and the neuron becomes depolarized. With a short delay, potassium channels open to let K^+ ions flow out of the cell while Na^+ channels start to close again, decreasing the positive charge in the cell and causing repolarisation.

The width of AP varies on the temperature, but in vertebrates is usually between $0.3ms$ and $2ms$ [Betts, 2013, Fohlmeister and Miller, 1997a, Pfützner, 2012, Rattay and Danner, 2014].

2.3.3 Refractory Period

During the AP, the neuron cannot be excited to create another AP. This is called the absolute refractory period. After the repolarisation phase passes the threshold, the Na^+ ion channels close and further stimulation becomes possible. This however, would require a higher stimulation energy in order to overcome the open K^+ ion channels and the time needed for the inactivation variable h to come back to its resting value is known as relative refractory period. At the end, when the membrane is back to its resting potential, the refractory period is over.

2.3.4 Graded Potential

The AP is used for information transmission by RGCs. Other cell types, like the bipolar cells and amacrine cells, have subtypes which do elicit AP and subtypes which do not [Baden et al., 2013]. Most sensory neurons, like photoreceptor cells, do not elicit action potentials. In general, if the neuronal packing is dense enough to have processes which are only a few $100\mu m$ long, a transmission of information without AP amplification is possible. This communication without AP works through graded potentials, thereby allowing transmission through the synapses of potentials below the threshold of other cells. Depending on whether the potential is inhibiting (hyperpolarizing) or excitatory (depolarizing), it can be distinguished between:

- **EPSP** (excitatory postsynaptic potential): positive potential
- **IPSP** (inhibitory postsynaptic potential): negative potential

The potential of all connected synapses will be summed (summation) in the dendritic tree of the postsynaptic neuron. If the result of the summation is sufficient to uplift the intracellular potential in the initial segment from approximately $-70mV$ (resting potential) to $-55mV$, the threshold of the neuron would have been reached, and an AP will be elicited [Betts, 2013].

2.4 Retinal Prosthesis

The perception of light can not only be caused by natural sense, but can also be generated by electrical impulses. This phenomenon is called phosphenes. All modern visual prosthesis are based on this effect. The only requirement is an intact optic pathway from the retina to the visual cortex, in a best-case scenario, without rewiring. Based on the location of the electrode array, the prosthesis can be differentiated between sub-retinal, epi-retinal and suprachoroidal.

Today, epi-retinal and sub-retinal prostheses achieve a comparable performance in clinical tests on human subjects, while suprachoroidal is in early-stage development [Zrenner, 2014].

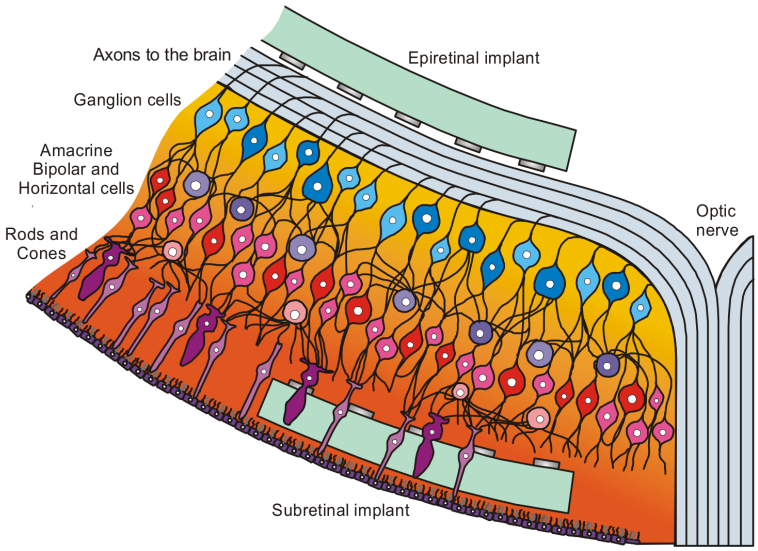


Figure 2.6: Possible implantation sites for a retinal prosthesis, adapted from Resatz [Resatz, 2005].

2.4.1 Sub-retinal Stimulation

The most advanced implant of sub-retinal stimulation is the prosthesis developed by Zrenner et al. in Tübingen, Germany. A sub-retinal implant is placed close to the region formerly occupied by photoreceptor cells. The main target neurons during sub-retinal stimulation are bipolar cells close to the stimulating electrodes. Therefore, a thin foil with electrode arrays is implanted under the retina. The residual visual pathway remains intact and processes the signal. The light-detecting photodiode can be placed directly on the implanted foil. These diodes record the incident light which passes through the lens and the iris of the eye. This technique does not require external cameras and allows correct interpretation of the eye movement and target finding. External amplification however, and thus a power source, is still required, because the current generated by the photodiodes is insufficient to create vision [Chow et al., 2004]. The implanting process is invasive, but it is not a surgical problem for trained surgeons [Zrenner et al., 2011]. Current applications are still for temporary use only, but allow restoring vision for daily tasks like recognizing pattern, reading $8cm$ big letters and identifying crockery. These results were achieved by an array of 1500 electrodes [Zrenner et al., 2011].

2.4.2 Epi-retinal Stimulation

A different approach directly stimulates the axons of the RGC on their retinal output pathway before they form the optical nerve. Especially when cells for image analysis are damaged by the disease suffered, epi-retinal stimulation can achieve vision restoration, while the other methods presented rely on these cells. To generate the pattern for perception, external image capturing (usually with a head-mounted camera) is required. The signals, generated by an external video processing unit, have to be converted to a variation of pulse pattern and stimulation intensity. Since each axon is carrying signals from many connected cells of presynaptic neurons (neurons of lower cell layers)- where different image-processing steps happen, implants with higher resolution would require more sophisticated image processing [Zrenner et al., 2011]. In present-day implants, the number of electrodes is small compared to sub-retinal stimulation. The two commercially developed prosthesis Argus[1] I (2004) and II (2009) have 16 and 60 electrodes, respectively. Consequently, the maximal resolution is lower than in sub-retinal implants with 1500 electrodes. The fixation of the implant, which is achieved by a tack, remains a problem. It can, for example, cause fibrosis around the direct area of the tack [Yue et al., 2015].

A remaining obstacle, which is also present in other approaches, is the selective stimulation of desired single cells or cell-groups, which are required to generate an image. Additionally, the electrode location does not necessarily correlate with the point of perception, since the RGC have different courses to the optical nerve. Chapter 4.3 analyses this problem in more detail.

2.4.3 Supra-choroidal Stimulation

A third, but less developed approach, is to place a micro-electrode-array between sclera and choroid. This method is less surgically invasive for the patient. The problem with supra-choroidal stimulation is the high resistivity of the pigmented epithelium and the comparably far distance to the targeted bipolar cells, which makes high currents necessary to achieve neuronal activation. Due to of a lack of clinical studies in this approach, it remains unclear if the spatial resolution is sufficient to restore vision. Like in the epi-retinal stimulation, external image capturing is necessary for this approach.

[1] http://www.secondsight.com/

2.5 Subtypes of Computational Retina Models

Its clear structure and the lack of feedback from the brain allow sophisticated modelling approaches of the retina. According to the categorisation of Guo and coworkers (2014), the models can be separated as follow (listed in order of increasing abstraction level): single-compartment models, morphologically-realistic models, block-compartment models, continuum models and discrete neuronal network models [Guo et al., 2014]. These different abstraction levels can be combined. Not all of these models are used in this thesis, but if a concept is applied, the corresponding chapter is mentioned.

2.5.1 Single-Compartment Models

Single-Compartment models describe specific points of a neuronal cell, usually derived from space clamp experiments. The models which are presented further on in this thesis: Fohlmeister & Miller and Hodgkin & Huxley, all belong to this group [Fohlmeister and Miller, 1997a, Hodgkin and Huxley, 1952]. They give a mathematical model for the current in a single compartment, but fail in more complex tasks like AP propagation. The analysis of membrane properties in Chapter 3 uses this type of model.

2.5.2 Morphologically-Realistic Models

Models of this type consider the morphological structure as well as the electrical properties of neighbouring regions. This is usually implemented by multiple interconnected compartments applying cable theory. The influence of the external potential, which can be calculated by the activating function, can also be applied in this model type. Later models of Fohlmeister and coworkers use detailed morphologies for the modelling and belong to this group [Fohlmeister et al., 2010, Fohlmeister and Miller, 1997b]. The neuron in Chapter 4 is modelled in a semi-morphologically-realistic way.

2.5.3 Block-Compartment Models

More complex structures of neurons make too detailed modelling impractical and computationally expensive. This approach tries to simplify each neuronal part like the soma and the axon, to one single compartment, each with homogeneous properties.

2.5.4 Continuum Models

Continuum models consider the retina as an infinite, homogeneous bulk volume. Single neurons or their properties are not considered anymore, hence their membrane behaviours are averaged and extrapolated to the entire volume. This model is suitable for simulating the current flowing in the extracellular space or across membranes. A common tool for filling the bulk material and applying the equations for the macro-property is the finite elements analysis. The first and still important continuum model was presented by Dokos and coworkers (2005), who modelled the active RGC layer [Dokos et al., 2005].

2.5.5 Block-Structured Models

The block model does not follow a biological approach, but tries (similar to electrical engineering) to find functional units according to statistical behaviour like filters and transformations. These units combined, can model effects from motion detection to vision. Since they are not biologically motivated, they are not suitable for finding conclusions on the neuronal level, but can give statements about the interaction of neuronal blocks, and lead to a better understanding of the connections in the visual system. Using this model, Keat and coworkers (2001) were able to create a model which is able to simulate a wide range of spike trains and different response types [Keat et al., 2001].

2.5.6 Discrete-Neuronal Network Models

Discrete-Neuronal network models try to use morphologically-realistic models to integrate neurons in high quantities in one model, in order to simulate retinal structures (or part of structures). The physiological accuracy can vary between different models, because structures which are not investigated can be left out. One of the largest retina models, the 'Virtual Retina' with 100.000 neurons, is part of this category and can process biologically plausible images [Wohrer and Kornprobst, 2009].

2.6 Basics of Electrical Stimulation

2.6.1 Electrical Network Model

In order to simulate the complex geometry of neurons, a mechanism based on an RC compartment model (resistor-capacitor model) is used to calculate the electrical properties in the neuron. This way, the neuron is divided into small compartments (usually between 1µm and 40µm) and each compartment is represented by an RC circuit. This allows calculation of the (static) voltage in the center of each compartment [Rattay, 1999].

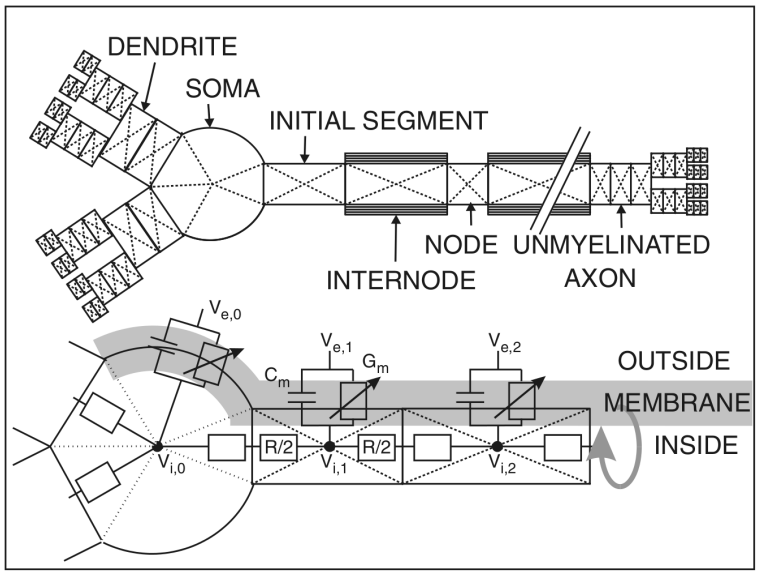


Figure 2.7: Equivalent circuit diagram of a neuron [Rattay et al., 2002]

V_e represents the external (extracellular) voltage, which can either be calculated by analytical methods like ideal point sources and idealized surface electrodes, or by numerical methods. These numerical methods are finite elements, finite differences, boundary element or Galerkin methods. Anyway, V_e is considered independent by the compartment. V_i is the mean value of the inner voltage in a specific compartment. C_m is the membrane capacity. R the axial (electrical) resistance of each compartment. G_m is the passive membrane conductance. In active models, the membrane current I_{ion} is calculated by gating mechanisms. Gating variables mathematically describe the complex interplay of opening and closing ion channels.

Because there are no current sources within compartments, the sum of all currents is zero, and the application of Kirchhoffs Law leads to:

$$\frac{d(V_{i,n} - V_{e,n})}{dt} \cdot C_{m,n} + I_{ion,n} + \left[\frac{V_{i,n} - V_{i,n-i}}{R_n/2 + R_{n-1}/2} + \frac{V_{i,n} - V_{i,n+i}}{R_n/2 + R_{n+1}/2} \right] = 0 \tag{2.3}$$

This representation where n is the compartment number allows us to see the three types of current in every single compartment:

Membrane Current

$I_c = \frac{d(V_{i,n} - V_{e,n})}{dt} \cdot C_{m,n}$ describes the current which is needed to load and unload the electrical capacity created by the cell membrane.

Ionic Current

Depending on the model, a passive approach $I_{ion} = g_m \cdot V$ or an active approach, for instance $I_{ion} = I_{Na} + I_K + I_{leak}$ by Hodgkin Huxley (Chapter 2.7.1), is used to calculate the ionic current. I_{ion} can consist of multiple ion channel types, as proposed in recent membrane models [Fohlmeister et al., 2010, Fohlmeister and Miller, 1997a].

Ohmic Current

Between each compartment and its neighbouring compartments arises a potential difference, which leads to an ohmic current over the axoplasmic resistance between the compartments: $\left(\frac{V_{i,n} - V_{i,n-i}}{R_n/2 + R_{n-1}/2} + \frac{V_{i,n} - V_{i,n+i}}{R_n/2 + R_{n+1}/2} \right)$

2.6.2 Activating Function

Taking into consideration that the reduced membrane voltage V (the potential difference between the interior and exterior of the membrane) is calculated by

$$V = V_i - V_e - V_{rest} \tag{2.4}$$

leads to a set of differential equations which describe the time dependence of the potential in every compartment n:

$$\frac{d(V_n)}{dt} = \frac{1}{C_{m,n}}\left[-I_{ion,n} + \frac{V_n - V_{n-i}}{R_{n-1}/2 + R_n/2} + \frac{V_{n+1} - V_n}{R_{n+1}/2 + R_n/2} + \dots \right.$$
$$\left. + \frac{V_{e,n-1} - V_{e,n}}{R_{n-1}/2 + R_n/2} + \frac{V_{e,n+1} - V_{e,n}}{R_{n+1}/2 + R_n/2} + \dots\right] \quad (2.5)$$

This allows the isolation of the extracellular voltage V_e, in order to set up an equation that describes the effect of a virtual stimulation current on a compartment n:

$$f_n = \frac{1}{C_{m,n}} \cdot \left[\frac{V_{e,n-1} - V_{e,n}}{R_{n-1}/2 + R_n/2} + \frac{V_{e,n+1} - V_{e,n}}{R_{n+1}/2 + R_n/2} + \dots\right] \quad (2.6)$$

In other words, when $V_i = V_{rest}$ (the neuron in resting state), the activating function f_n describes the voltage change of the membrane of one compartment. This is used to calculate the stimulating influence of an external electric field on the potential of each neuron [Rattay, 1999].

2.7 Membrane Models

2.7.1 Hodgkin Huxley Model (1952)

Alan Lloyd Hodgkin and Andrew Fielding Huxley found in 1952, in experiments with the nonmyelinated axons of the giant squid, a mathematical description for the currents passing through a cell membrane [Hodgkin and Huxley, 1952]. The squid axon was used because of its extremely large diameter. This was important to perform the space clamp method, where electrodes are inserted into the cell. This finding describes the total ionic current as a sum of 3 single currents carried by sodium ions I_{Na}, potassium ions I_K, and a leakage current I_l made up of chloride and other ions.

$$I_{ion} = I_{Na} + I_K + I_{leak} \quad (2.7)$$

These currents arise from different ions inside and outside of the cell membrane. Electrically speaking, the cell membrane itself acts as a capacitor while the ion channels, because of their active ion transportation, act as a battery or current source. [Gerstner and Kistler, 2002]

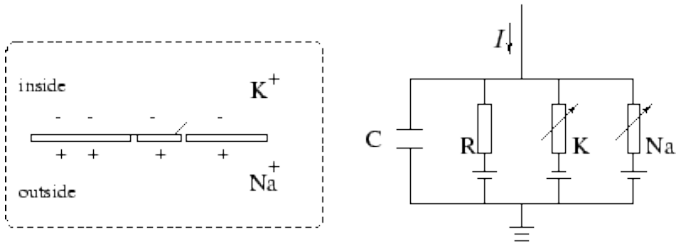


Figure 2.8: Schematic diagram for the Hodgkin-Huxley model [Gerstner and Kistler, 2002].

Each of these currents is expressed by a conductance g, the membrane potential E and a specific ionic equilibrium potential E_{Na}, E_K, E_l. E is the absolute potential and V is a relative potential, relative to the resting potential. This leads to the well known form of the Hodgkin Huxley equation for ion currents:

$$I_{Na} = \bar{g}_{Na} \cdot m^3 \cdot h \cdot (V - V_{Na}) \tag{2.8}$$

$$I_K = \bar{g}_K \cdot n^4 \cdot (V - V_K) \tag{2.9}$$

$$I_l = \bar{g}_l (V - V_l) \tag{2.10}$$

$\bar{g}_{Na}$ $\bar{g}_K$ and $\bar{g}_l$ are a constant for conductance per cm^2. The so-called gating variables n, m and h are statistical (dimensionless) variables between 0 and 1 and evolve according to these differential variables [Rattay et al., 2002]:

$$\frac{dm}{dt} = [-(\alpha_m + \beta_m) \cdot m + \alpha_m] \cdot k \tag{2.11}$$

$$\frac{dh}{dt} = [-(\alpha_h + \beta_h) \cdot h + \alpha_h] \cdot k \tag{2.12}$$

$$\frac{dn}{dt} = [-(\alpha_n + \beta_n) \cdot n + \alpha_n] \cdot k \tag{2.13}$$

$$k = 3^{0.1(T-0.63)} \tag{2.14}$$

Figure 2.9 shows the steady state gating variables presented as a function of voltage. The inhibiting character of h can be seen very clearly, contrary to the exciting character of m and n ion gating variables.

T represents the temperature. The original temperature was 6.3 ℃ and a temperature coefficient k for temperatures up to 31.2 ℃ was introduced.

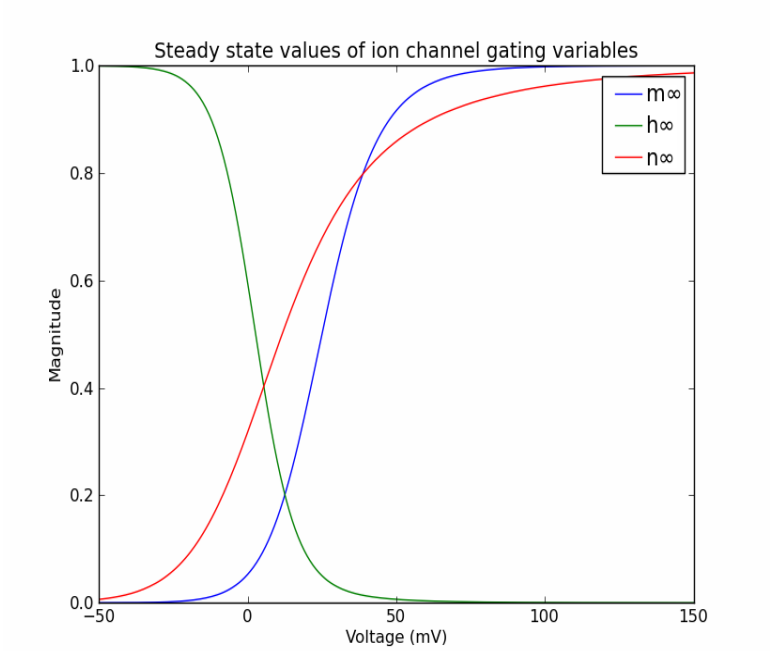


Figure 2.9: The two sodium (m, h) and one potassium (n) variables, adapted from Galbraith [Galbraith, 2011].

At higher temperatures, a so-called heat block will set in, which means the axon stops AP propagation [Rattay et al., 2002]. α and β are called rate constants and are gained by fitting curves on experimental data. α describes the transfer rate of ions from outside, in and β is the transfer from inside, out [Hodgkin and Huxley, 1952].

Including the capacity property of the membrane, we get the total membrane current density i as a function of time and voltage, which is according to Kirchhoff's laws 0:

$$i = c_m \cdot \frac{d(V)}{dt} + \bar{g}_K \cdot n^4 \cdot (V - V_K) + \bar{g}_{Na} \cdot m^3 \cdot h \cdot (V - V_{Na}) + \bar{g}_l \cdot (V - V_l) = 0 \tag{2.15}$$

For applications where an intracellular stimulation is simulated, a stimulus current in the cell is needed. In this case, a stimulation current I_{stim} is added to the existing currents (ionic current and the capacitive current) [Rattay et al., 2002]

$$I = I_c + I_{ion} + I_{leak} + I_{stim} \tag{2.16}$$

$$\frac{d(V)}{dt} = \left[-\bar{g}_{Na} \cdot m^3 \cdot h \cdot (V - V_{Na}) - \bar{g}_K \cdot n^4 \cdot (V - V_K) - \bar{g}_l \cdot (V - V_l) + i_{stim} \right] / c_m \tag{2.17}$$

Equation 2.17 describes the membrane voltage over time for an active single compartment model in current clamp mode. Figure 2.10 shows two action potentials during a 25ms, $10\mu A/cm^2$ pulse.

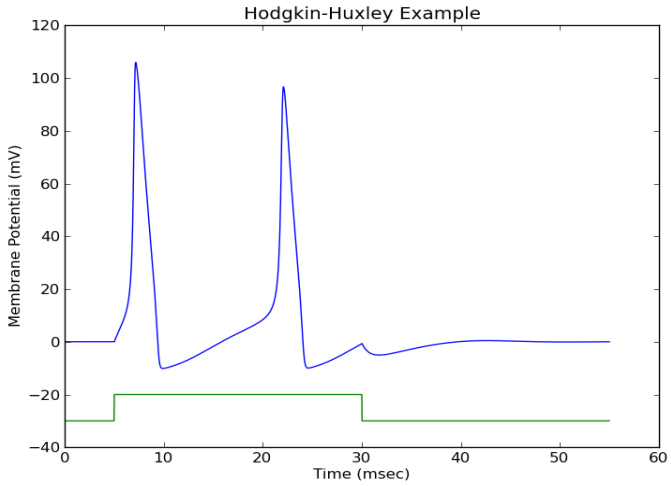


Figure 2.10: $25ms$ and $10\mu A/cm^2$ input pulse (green), noticeable Hodgkin Huxley spikes (blue) with refractory period between the spikes [Galbraith, 2011]

2.7.2 Fohlmeister and Miller Model (1997)

Jurgen Fohlmeister and Robert Miller developed in their space-clamp experiments on retinal ganglion cells (RGC) of the tiger salamander, an extension to the gating mechanism model of Hodgkin and Huxley (Chapter 2.7.1). This model consists of 5 non-linear ion channel types which gate the currents $I_{Na}, I_K, I_{Ca}, I_{K,A}$ and $I_{K,Ca}$, and 1 linear leakage channel I_L. While this model is one of the most important models for retinal simulation, it is derived from amphibian tissue and is not fully representative of the human retina. The basic concept of the membrane current consisting of these 5 ionic currents was already published by Lipton and Tauck in 1987 [Lipton and Tauck, 1987].

Sodium Current

The non-linear sodium channels are voltage gated and consist of activating α_m and β_m and inactivating α_h and β_h channels. The corresponding I_{Na} is the fastest reacting current and is calculated by:

$$i_{Na} = \bar{g}_{Na} \cdot m^3 \cdot h \cdot (V - V_{Na}) \tag{2.18}$$

V_{Na} and $\bar{g}_{Na}$ are, like in Hodgkin & Huxley, the constant equilibrium potential $V_{Na} = +35mV$ for sodium and channel density $\bar{g}_{Na}$ 50 mS/cm^2.

Delayed Rectifier Potassium Current

Being activated by the influence of Na, the I_K (or delayed rectifier) might also play a role in stabilizing the behaviour of the cell, without any inactivating component. The equilibrium potential V_K of potassium is $-75mV$, while channel density $\bar{g}_K$ is 12 mS/cm^2.

$$i_K = \bar{g}_K \cdot n^4 \cdot (V - V_K) \tag{2.19}$$

A-type Potassium Current

A transient A-type (rapidly inactivating) potassium current $I_{K,A}$ is calculated by:

$$i_{K,A} = \bar{g}_A \cdot a^3 \cdot h_A \cdot (V - V_K) \tag{2.20}$$

$I_{K,A}$, with its activating α_A and β_A and inactivating α_{hA} and β_{hA} channels, plays an important role in stabilizing the cell (i.e. inhibits spontaneous firing). It behaves similarly to a delayed rectifier, but with an additional inactivating behaviour. The corresponding potassium activated channels have a channel density $\bar{g}_A$ of 36 mS/cm^2 [Fohlmeister and Miller, 1997a].

Calcium Current

The voltage-dependent calcium current I_{Ca} is gated without inactivating channels and plays an important role in controlling impulse frequency, by removing the cytoplasmic calcium. This is caused by the lack of inactivation kinetics, therefore the reduction of calcium in the cell happens through the concentration difference between inner and outer cell after an action potential. The current is driven by calcium channels ($\bar{g}_{Ca} = 2.2mS/cm^2$) α_c and

β_c. The Ca channels have no inactivating kinetics, but appear inactivating because of increasing intracellular calcium concentration.

$$i_{Ca} = \bar{g}_{Ca} \cdot c^3 \cdot (V - V_{Ca}) \tag{2.21}$$

As previously mentioned, V_{Ca} is not defined as an equilibrium potential constant like the other potentials, but modelled according to the Nernst equation (Equation 2.22). R is the gas constant ($8.31441\ J \cdot K^{-1} \cdot mol^{-1}$), T is the temperature in Kelvin and in this case 295.15 K (22℃), z is valence and in case of $[Ca^{2+}]$ 2, F is the Faraday constant (96485 C/mol), $[Ca^{2+}]_i$ is the internal calcium concentration, derived by equation 2.26 and $[Ca^{2+}]_e$ is the external calcium concentration (fixed to 1.8 mM) [Rattay et al., 2002].

$$V_{Ca} = \frac{RT}{zF} \cdot ln \frac{[Ca^{2+}]_e}{[Ca^{2}+]_i} \tag{2.22}$$

Ca-Activated Potassium Current

Calcium activated potassium current $I_{K,Ca}$ is modelled to respond to influx through the calcium channels. This type of potassium channel is not directly dependent on voltage and has only activating characteristics.

$$i_{K,Ca} = g_{K,Ca} \cdot (V - V_K) \tag{2.23}$$

The channel density $g_{K,Ca}$ is, contrary to the other gating types, not constant, but derived from the ionic concentration in the RGC. It depends on the influx of Ca^{2+} in the cell, the cell size and changes in the internal calcium Ca_i^{2+} concentration [Fohlmeister and Miller, 1997a] :

$$g_{K,Ca} = \bar{g}_{K,Ca} \cdot \frac{([Ca^{2+}]_i / Ca_{diss}^{2+})^2}{1 + ([Ca^{2+}]_i / Ca_{diss}^{2+})^2} \tag{2.24}$$

$$\bar{g}_{K,Ca} = 0.05\ mS/cm^2 \tag{2.25}$$

Ca_{diss}^{2+} is the calcium dissociation constant $10^{-6}M(mol/dm^3)$. $[Ca^{2+}]_i$ describes the inner cellular Ca^{2+} concentration and is, in case of no Ca^{2+} influx, equal to the residual concentration $[Ca^{2+}]_{res} = 10^{-7}M$, otherwise it can be calculated (for spherical compartments with the radius r) [Fohlmeister et al., 1990, Fohlmeister and Miller, 1997a]:

$$\frac{d[Ca^{2+}]_i}{dt} = \frac{3}{r} \cdot \frac{-i_{Ca}}{2F} - \frac{[Ca^{2+}]_i - [Ca^{2+}]_{res}}{\tau_{Ca}} \tag{2.26}$$

$\frac{3}{r}$ is the surface-area-to-volume ratio of a sphere. For compartment shapes other than spherical compartments, a generic form of Equation 2.26 can be used, where A is the surface area and V the volume of the compartment:

$$\frac{d[Ca^{2+}]_i}{dt} = \frac{A}{V} \cdot \frac{-i_{Ca}}{2F} - \frac{[Ca^{2+}]_i - [Ca^{2+}]_{res}}{\tau_{Ca}} \tag{2.27}$$

The calcium time constant is set in all Fohlmeister models to $\tau_{Ca} = 1.5ms$. $I_{K,Ca}$ has been suspected of playing a role in stabilizing the behaviour of the cell, controlling the impulse frequency and determining the duration of the action potentials [Fohlmeister and Miller, 1997a].

Capacitive Current (Membrane Current)

The capacitive current or cell membrane current is modelled like in Hodgkin and Huxley (Figure 2.8) and uses a membrane capacity $c_m = 1\mu F/cm^2$, which has been found in neuronal cells of squid, mice and other vertebrae [Gentet et al., 2000].

$$I_C = C_m \cdot \frac{d(V)}{dt} \tag{2.28}$$

Leakage Current

The leakage current i_L is the only linear leakage current in this model and represents the input resistance of the stimulating input. The constant equilibrium potential $V_L = -60mV$ and channel density $\bar{g}_L = 1\ mS/cm^2$ vary on different studies. The input resistance R_n of $1G\Omega$ has been assumed.

$$I_L = \bar{g}_L(V - V_L) \tag{2.29}$$

Ion Channel Stimulation

Like in the Hodgkin Huxley case, the 5 ionic currents and the membrane current follow Kirchoff's law and equal the stimulating current if present, otherwise being zero:

$$I_{stim} = C_m \cdot \frac{d(V)}{dt} + I_{Na} + I_{Ca} + I_K + I_{K,A} + I_{K,Ca} + I_L \tag{2.30}$$

Gating Dynamics

Due to lack of detailed knowledge about the gating dynamics of membrane channel molecules, the gating stoichiometry is modelled by the variables m, h, c, n, h_A and a (in equation 2.31 mentioned as x), according to the first order kinetic equation by Hodgkin and Huxley.

$$\frac{dx}{dt} = [-(\alpha_x + \beta_x) \cdot x + \alpha_x] \tag{2.31}$$

The resting state for a gating variable can be computed by:

$$x_0 = \frac{\alpha_x}{(\alpha_x + \beta_x)} \tag{2.32}$$

The detailed parameter (α and β for each channel type) were scaled to fit the experimental data and can be found in Table 2.1.

channel type	α	β
Na^+ channel activation	$\alpha_m = \frac{-0.6 \cdot (E+30)}{e^{-0.1 \cdot (E+30)} - 1}$	$\beta_m = 20 \cdot e^{-(E+55)/18}$
Na^+ channel inactivation	$\alpha_h = 0.4 \cdot e^{-(E+50)/20}$	$\beta_h = \frac{6}{e^{-0.1 \cdot (E+20)} + 1}$
Ca^{2+} channel activation	$\alpha_c = \frac{-0.3 \cdot (E+13)}{e^{-0.1 \cdot (E+13)} - 1}$	$\beta_c = 10 \cdot e^{-(E+38)/18}$
K^+ channel activation	$\alpha_n = \frac{-0.02 \cdot (E+40)}{e^{-0.1 \cdot (E+40)} - 1}$	$\beta_n = 0.4 \cdot e^{-(E+50)/80}$
A channel activation	$\alpha_A = \frac{-0.006 \cdot (E+90)}{e^{-0.1 \cdot (E+90)} - 1}$	$\beta_A = 0.1 \cdot e^{-(E+30)/10}$
A channel inactivation	$\alpha_{A_h} = 0.04 \cdot e^{-\cdot (E+70)/20}$	$\beta_{A_h} = \frac{0.6}{e^{-0.1 \cdot (E+40)} + 1}$

Table 2.1: Rate constants for Na, Ca, K and A channels of Fohlmeister et al. 1997 [Fohlmeister and Miller, 1997a].

2.7.3 Fohlmeister, Cohen and Newman Model (2010)

Further research of Fohlmeister, Cohen & Newman regarding the spiking behaviour of rat and cat RGCs at different temperatures enabled the gathering of more detailed information of ion channels in mammals. Different parts of the retinal cell consist of a distinct density and distribution of ion channels. Apart from the ion channels, the temperature[2] was also found to influence the electric behaviour of ganglion cells in a non-linear way [Fohlmeister et al., 2010].

This model can be assumed as a modification of the Fohlmeister & Miller Model (Chapter 2.7.2) and therefore, it consists of the same five channel currents, namely:

$$I_{Na}, I_{Ca}, I_K, I_{K,A}, I_{K,Ca} \tag{2.33}$$

$I_{K,A}$, however, was found to be very small in comparison to I_K and was therefore excluded ($I_{K,A} = 0$). I_{Ca} was also found to be very small, but plays an important role in balancing the potassium current at the falling action potential.

The mathematical structure of the currents is equal to Fohlmeister & Miller Model (1997) 2.7.2 and described by the equations 2.18 (I_{Na}), 2.19 (I_K), 2.21 - 2.22 (I_{Ca}). The calcium activated potassium current $I_{K,Ca}$ is gated like in equations 2.24 to 2.26, because of the more complex calcium equilibrium. In contrast to Hodgkin & Huxley, in both Fohlmeister Models (Fohlmeister & Miller (1997) and Fohlmeister Cohen Newman (2010)), there is almost no temporal overlap at sodium and potassium currents (Figure 2.11).

Again, in the absence of any other current sources and sinks, the electrotonic current can be calculated by applying Kirchoff's law:

$$I_{stim} = C_m \cdot \frac{d(V)}{dt} + I_{Na} + I_{Ca} + I_K + I_{K,Ca} + I_L \tag{2.34}$$

The equilibrium potentials and the channel densities are different from the previous model. The channel densities depend as well, on the region of the cell, but remain unchanged througout wide temperature ranges.

[2]This thesis is especially focused on comparability to Fohlmeister & Miller model 1997 and Hodgkin & Huxley model, and therefore uses a temperature between 20°C and 23°C .

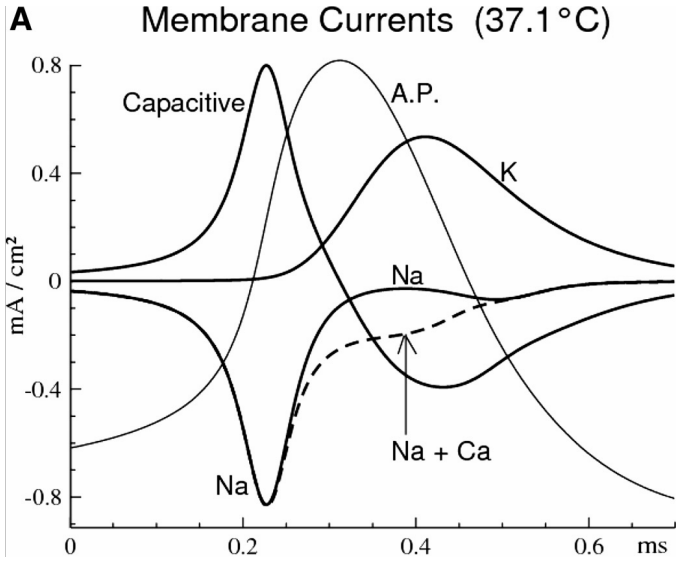


Figure 2.11: Estimated membrane currents during action potential [Fohlmeister et al., 2010]

	cat			rat		
	$\bar{g}_{Na}$	$\bar{g}_K$	$\bar{g}_{Ca}$	$\bar{g}_{Na}$	$\bar{g}_K$	$\bar{g}_{Ca}$
Soma	69.40	32	1.39	72.00	50.40	1.20
initial segment / axion hillock	100	50.10	0.836	141.10	67.80	0.753
Axon	124	50	04	124	50	0

Table 2.2: G-bar conductances $[mS/cm^2]$ for different channel types (Cat Beta, Rat Type I) and different neuron parts (Fohlmeister et al. 2010) [Fohlmeister et al., 2010].

	cat			rat		
	$\bar{g}_{Na}$	$\bar{g}_K$	$\bar{g}_{Ca}$	$\bar{g}_{Na}$	$\bar{g}_K$	$\bar{g}_{Ca}$
Sodium Band	1240	50	1.5	1240	50	1.5

Table 2.3: Modified G-bar conductances $[mS/cm^2]$ for sodium channel band (Cat Beta, Rat Type I). Sodium conductance has been increased 10-fold, compared to Table 2.2. [Fohlmeister et al., 2010, Jeng et al., 2011].

V_{Na}	V_K	V_L
58.34	-97.56	-62.17

Table 2.4: Equilibrium potentials [mV] for different channel types and 23.5℃ [Fohlmeister et al., 2010].

A leakage current I_L was also found, but with $0.1 mS/cm^2$ ($10.000 \Omega/cm^2$), only a relatively small effect was measured.

The gating kinetics as well as the calculation of the resting state, is equal to the gating Hodgkin Huxley and to Fohlmeister et al. 1997 (Equations 2.31 and 2.32). The structure of the standard gating rates α_x and β_x is equal, but the parameters are different from the gating rate constants compared to Fohlmeister et al. 1997, see Table 2.5.

channel type	α	β
Na^+ channel activation	$\alpha_m = \frac{-2.804 \cdot 0.463 \cdot (V+35)}{e^{-0.1 \cdot (V+35)} - 1}$	$\beta_m = 93.46 \cdot 0.463 \cdot e^{-(V+60)/18}$
Na^+ channel inactivation	$\alpha_h = 1.869 \cdot 0.463 \cdot e^{-(V+55)/20}$	$\beta_h = \frac{28.04 \cdot 0.463}{e^{-0.1 \cdot (V+25)} + 1}$
Ca^{2+} channel activation	$\alpha_c = \frac{-1.4 \cdot 0.463 \cdot (V+15)}{e^{-0.1 \cdot (V+15)} - 1}$	$\beta_c = 46.68 \cdot 0.463 \cdot e^{-(V+40)/18}$
K^+ channel activation	$\alpha_n = \frac{-0.0984 \cdot 0.478 \cdot (V+32.5)}{e^{-0.1 \cdot (V+32.5)} - 1}$	$\beta_n = 1.969 \cdot 0.478 \cdot e^{-(V+58.5)/76}$

Table 2.5: Rate constants for Na^+, Ca^{2+}, and K^+ channels for temperatures of 23.5℃. The "Kinetic Q_{10} Factors" are for Na^+ and Ca^{2+} 0.463 and for K^+ 0.478, respectively [Fohlmeister et al., 2010]

Chapter 3

Comparison of Membrane Properties

3.1 Ionic Current Flow

The membrane current and current density of an intracellular stimulated cylindrical soma were simulated. Additionally, these simulated currents were integrated over time to determine the absolute charge and relative charge density of an AP through the ionic channels of a neuron. For the simulations, the soma was chosen, because the Fohlmeister model was originally developed to simulate somas and, then extended for other neuronal parts. The simulated soma has a radius of $5\mu m$ (diameter $10\mu m$) and a length of $30\mu m$ (which leads to a surface area of $1099.56\mu m^2$, including lateral surfaces) and was simulated using a single compartment model. By convention, the inflow of current is marked by a negative sign, while the outflow has a positive sign.

In preparation for the current flow experiments, the minimal stimulation threshold was evaluated. A minimal stimulation threshold is the single current density which, when injected in the neuron for a stimulation time of $0.5ms$, initializes an AP within $5ms$ (listed in Table 3.1a)). If not explicitly stated otherwise, the double of the highest i_{min} has been used, in order to assure a spike without low-stimulus artefacts. A higher stimulation current density in the stimulus has little effect, because the membrane dynamics causes the ionic currents after the stimulus overcomes the threshold. For instance, Hodgkin-Huxley model stimulated at a threshold level of $36\mu A/cm^2$ instead of $118\mu A/cm^2$, increases the sodium-charge density from $-338.34nC/cm^2$ to $-338.98nC/cm^2$, which is much less than 1%.

	Fohlmeister 1997	Fohlmeister 2010	Hodgkin-Huxley
a) $i_{min}[\mu A/cm^2]$	59	53	36
b) $i_{stim}[\mu A/cm^2]$	118	118	118
c) $i_{stimDP}[\mu A/cm^2]$	177	159/236	/

Table 3.1: a) Minimal current density to initiate an AP by a $0.5ms$ pulse in soma of a specific membrane. b) stimulated current density with a $0.5ms$ pulse for power consumption experiments, c) stimulated current density with a $0.5ms$ pulse for double-pulse experiment (Chapter 3.2)

The simulation was performed at 20℃ for Hodgkin & Huxley and Fohlmeister et al. 1997 model and 23℃ Fohlmeister et al. 2010 model, respectively, to obtain comparable results between amphibian and mammalian electrophysiology. The HH model does not generate APs at mammalian body temperature (heat block) [Fohlmeister et al., 2010, Fohlmeister and Miller, 1997a, Greenberg et al., 1999, Hodgkin and Huxley, 1952].

	Fohlmeister 1997	Fohlmeister 2010	Hodgkin-Huxley
Na	-79.042 (-0.869)	-123.826 (-1.551)	-338.98 (-1.362)
K	143.056 (1.573)	191.846 (2.612)	374.73 (2.110)
K(A)	75.750 (0.833)		
K(Ca)	3.103 (0.034)	0.270 (0.003)	
Ca	-119.556 (-1.316)	-36.823 (-0.405)	
Leak	3.729 (0.040)	3.233 (0.035)	-10.72 (-0.11785)

Table 3.2: Ion charge density through the membrane, used up by the neuron to fire an AP in $[nC/cm^2]$. Absolute charge for this specific neuron in parenthesis $[pC]$.

The results in Table 3.2 show an increased energy efficiency in higher developed neurons. The cat soma, simulated by Fohlmeister et al. 2010 model, needs therefore only 50.86% of the energy to spike an action potential (AP), than the total energy used by the squid axon, modeled by Hodgkin and Huxley [Fohlmeister et al., 2010, Hodgkin and Huxley, 1952]. This behaviour can be explained by the simultaneous (activating) sodium current and (deactivating) potassium current in Hodgkin & Huxley membrane: the sodium current has to overcome the counteracting potassium current in order to spike an AP and has to remain high during the AP (Figure 3.2).

In Fohlmeister et al. 2010 (Figure 3.4), the potassium current starts to work against the sodium current with a much higher delay, which reduces the time because these currents are counterparts, therefore increasing the energy efficiency. All currents found to be monophasic, so they kept their polarity during the impulse.

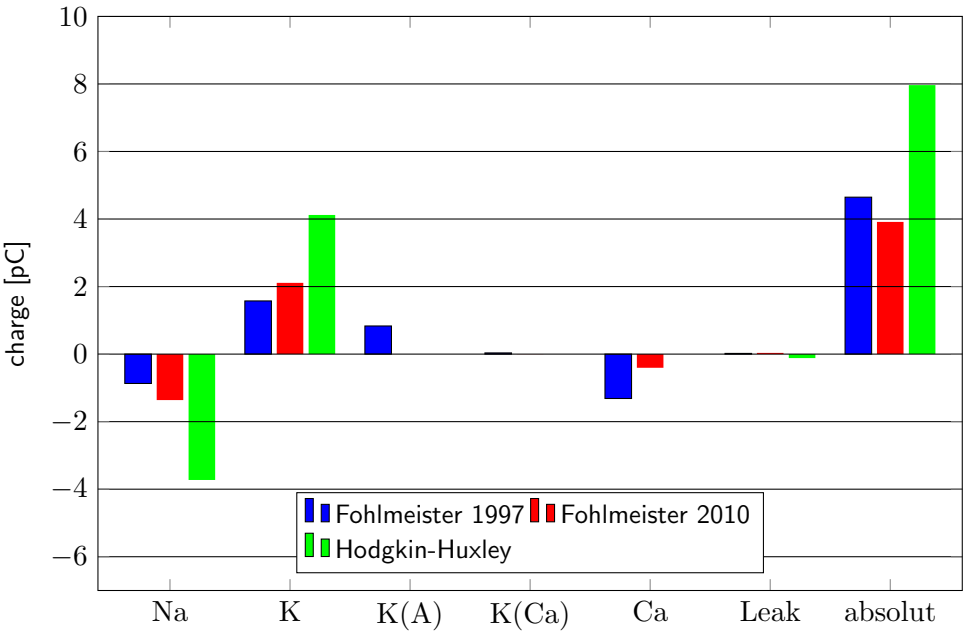


Figure 3.1: Power consumption comparison of individual ion types in different membrane models. Stimulation current density of $i = 118\mu A/cm^2$ (Table 3.1 b))

An interesting finding is that the more sophisticated cat soma from Fohlmeister et al. 2010 has a more increased energy demand on sodium (Na) and potassium (K) channels than the tiger-salamandar soma in Fohlmeister et al. 1997, despite the low sodium conductance of $g_{Na} = 39.29mS/cm^2$ in the cat-cell, compared to the tiger salamanders $g_{Na} = 50mS/cm^2$ [Fohlmeister et al., 2010, Fohlmeister and Miller, 1997a]. This increased energy consumption is compensated by a reduction of the calcium current and the absence of an A-type potassium current ($i_{K(A)}$) in the cat soma. These optimizations in the membrane lead to 15.76% overall improvement of energy efficiency of the feline neuron, compared to the tiger salamandar soma. This energy efficiency advantage might allow for a faster reaction time of the more developed cat soma (Figure 3.3 and 3.4).

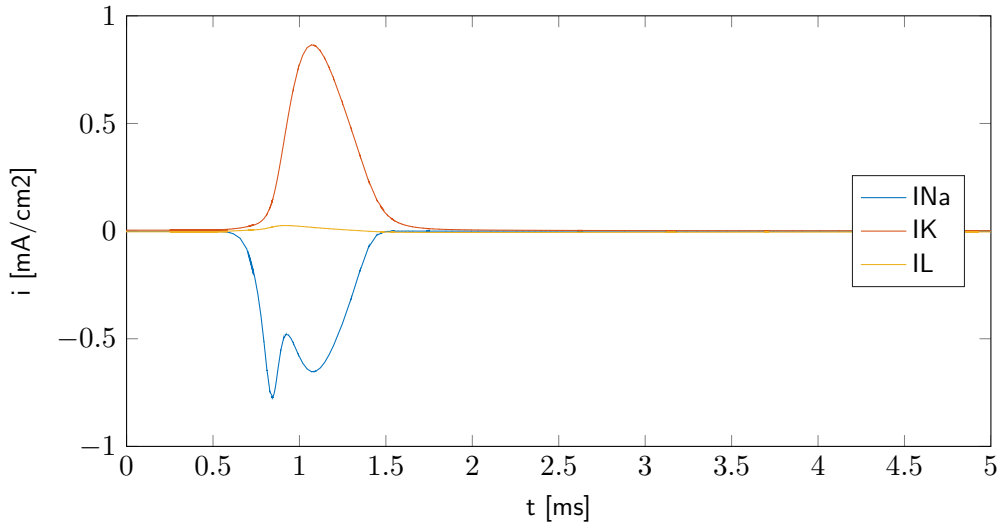


Figure 3.2: Current diagram of Hodgkin and Huxley [Hodgkin and Huxley, 1952]: Intercellular stimulation with a stimulation. The negative I_{Na} double-peak is not an artefact, but a typical behaviour for Hodgkin-Huxley. Model properties: Morphology: Single compartment; Channels: Hodgkin and Huxley; Temperature: 20°C; Pulse: monophasic $0.2ms$; Stimulus current density: $i = 118\mu A/cm^2$

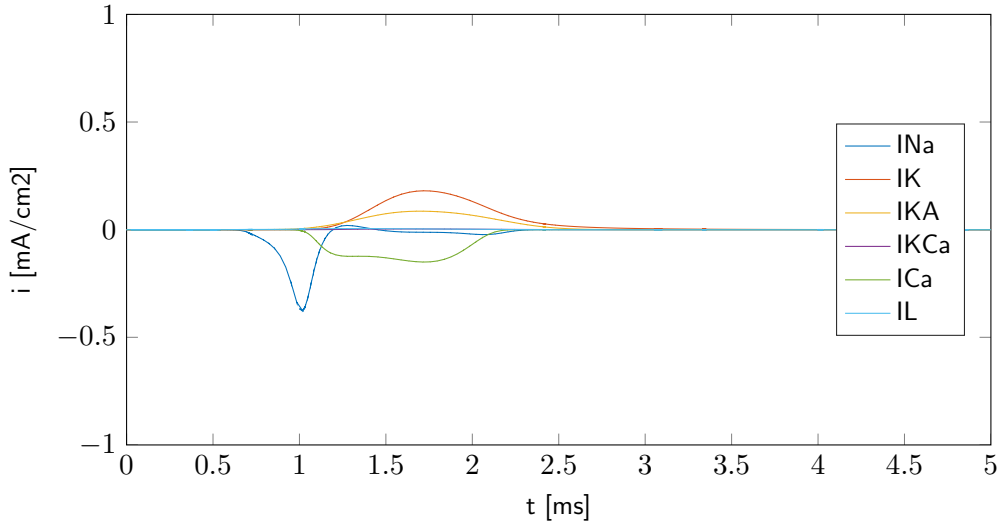


Figure 3.3: Current diagram of Fohlmeister et al. 1997 [Fohlmeister and Miller, 1997a]. Model properties: Morphology: Single compartment; Channels: Fohlmeister et al. 1997 ($g_l = 0.05mS/cm^2$); Temperature: 20°C; Pulse: monophasic $0.2ms$; Stimulus current density: $i = 118\mu A/cm^2$

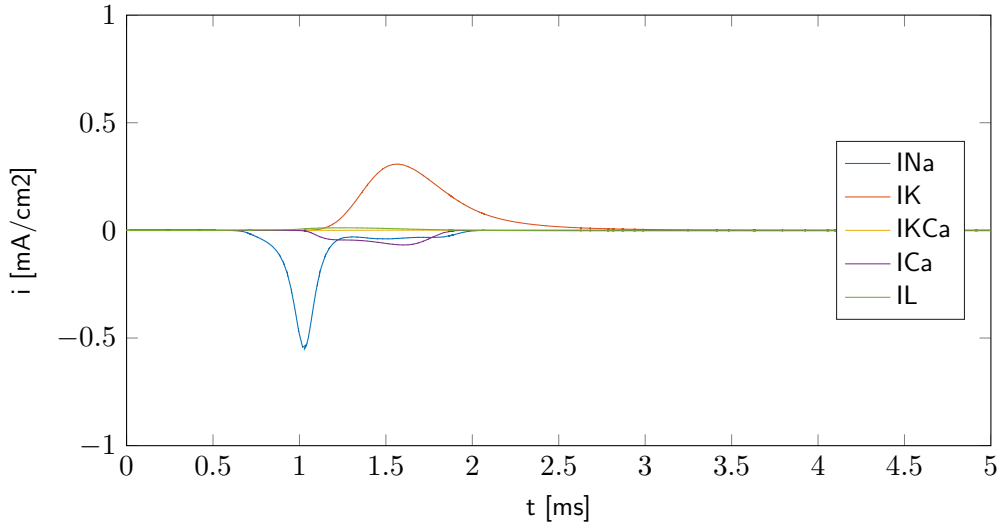


Figure 3.4: Current diagram of Fohlmeister et al. 2010 [Fohlmeister et al., 2010]. Morphology: Single compartment; Channels: Fohlmeister et al. 2010; Temperature: 23°C; Pulse: monophasic $0.2ms$; Stimulus current density: $i = 118\mu A/cm^2$

3.2 Double-Pulse Experiment

Methods

To compare the inter-peak times of both Fohlmeister models in a soma, a double-pulse experiment with a variation of inter-pulse time was performed. The stimulation in both models was applied in an intra-cellular manner, in a single compartment, and with a current density of triple the minimal threshold, which is $177\mu A/cm^2$ for Fohlmeister et al. 1997 and $159\mu A/cm^2$ for Fohlmeister et al. 2010 (Table 3.1c)). Additionally, as a third measurement, the more sensitive Fohlmeister et al. 2010 was stimulated by 4-fold threshold $236\mu A/cm^2$. The factor 3 is necessary to keep the spiking delay low and the second spike within the parameters of this simulation. Fohlmeister et al 2010 does not show a second spike within the following parameters when stimulated with a current density of below $115\mu A/cm^2$ ($4ms$ inter-pulse time).

An AP was counted as successful stimulation if membrane voltage crossed $0mV$ within $10ms$ after start of the simulation. The inter-peak time was measured on the voltage peak of each AP [Fohlmeister et al., 2010, Fohlmeister and Miller, 1997a].

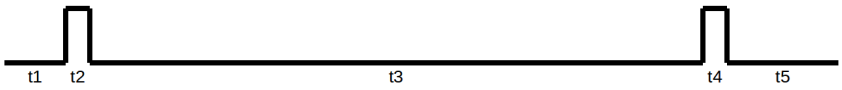


Figure 3.5: Input function for intracellular stimulation: $t_1 = 0.5ms$ is the starting time, $t_2 = 0.2ms$ is the first pulse, t_3 is the inter-pulse-time, $t_4 = 0.2ms$ is the second pulse and t_5 is the cool-down time, until end of simulation. t_3 and t_5 are not in scale; $t_1+t_2+t_3+t_4+t_5 = 10ms$ (constant).

Inter-pulse time

Figure 3.6 shows the relation between inter-peak-time (i.e. duration between two APs) and inter-pulse time (i.e. t_3 in Figure 3.5). At higher inter-pulse times $t_3 > 3ms$, the curve ascends at an angle of 45° because after the refractory period, the two AP do not affect each other. At shorter inter-pulse times $t_3 < 3ms$, the second AP interferes with the refractory time of the first AP, which leads to different behaviours of the three depicted simulations.

At inter-peak times below $3ms$, the Fohlmeister et al. 2010 model shows quite a surprising effect of a second pulse, which does not answer the expec-

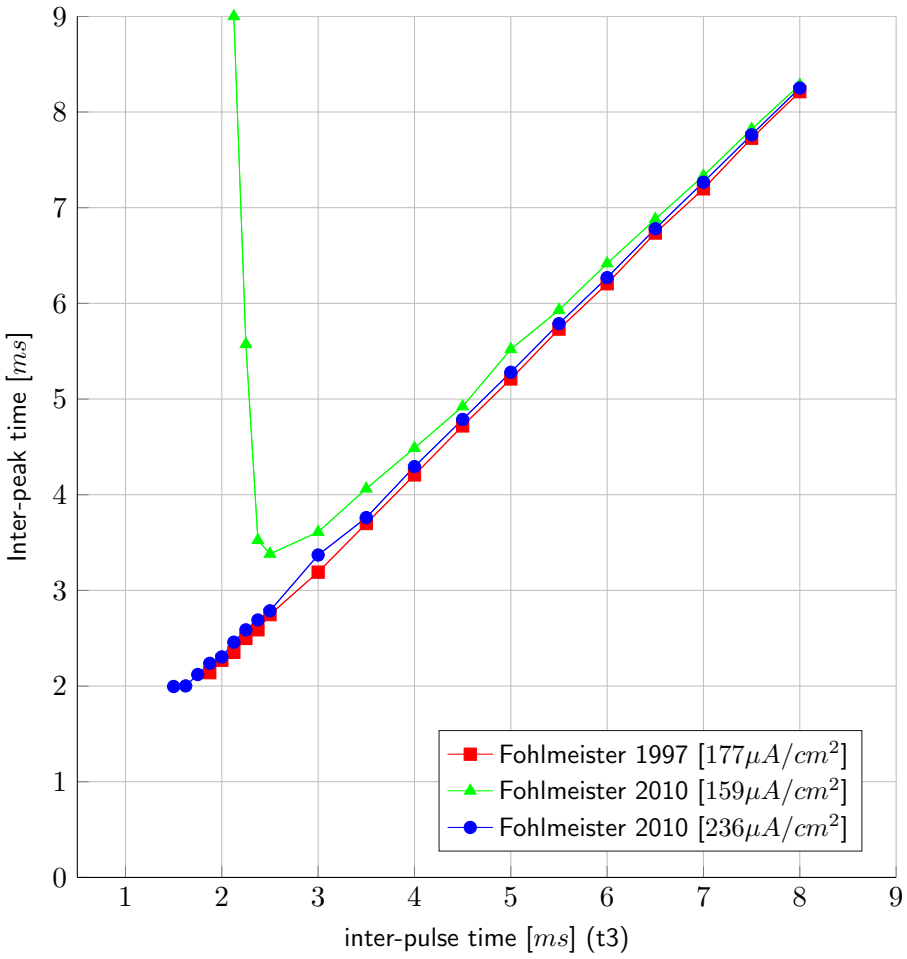


Figure 3.6: Relation of inter-peak-time to inter-pulse-time.

tations in time behaviour, but can be explained by charging the capacitive part of the membrane during the relative refractory time after the first spike. This charge of the second pulse reduces the time needed for self-stimulation, and therefore the second spike which can be seen after $2ms$, is an early self-stimulation. Self-stimulation happens in a $236\mu A/cm^2$ stimulated or unstimulated Fohlmeister et al. 2010 as well, but not within the simulation time of $10ms$.

Furthermore, it can be seen that in Fohlmeister et al. 2010 a higher stimulation pulse ($236\mu A/cm^2$) can overcome the relative refractory time from $1.5ms$ inter-pulse time on, similar to the Fohlmeister et al. 1997. Please note that the Fohlmeister et al. 1997 is stimulated much closer to the minimal threshold and a second AP is not elicited when stimulated

in refractory time. Fohlmeister et al. 1997 however, does not show self-stimulation. The AP caused by the three different models is illustrated at an inter-pulse time of 2.25ms (Figure 3.7) and 4ms (Figure 3.8). Fohlmeister et al. 1997 has a more menial peak than Fohlmeister et al. 2010, which is a side-effect of the already shown higher energy efficiency [Fohlmeister et al., 2010, Fohlmeister and Miller, 1997a].

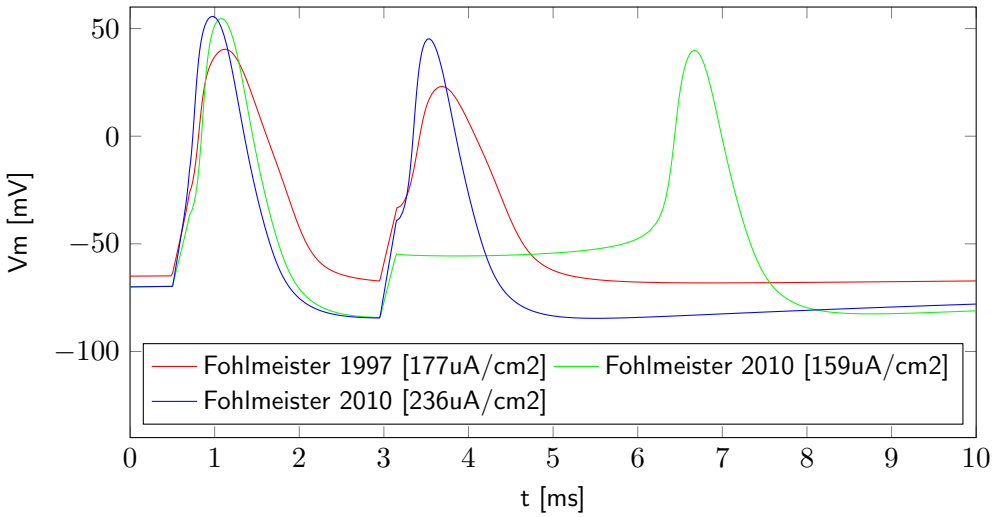


Figure 3.7: Comparison of the inter-peak-time of Fohlmeister et al. 2010 and Fohlmeister et al. 1997 [Fohlmeister et al., 2010, Fohlmeister and Miller, 1997a] with 2 stimuli and an inter-pulse-time t3=2.25ms. Model properties: Morphology: Single compartment; Channels: Fohlmeister et al. 2010 and Fohlmeister et al. 1997; Temperature: Fohlmeister et al. 1997: 20°C/ Fohlmeister et al. 2010: 23°C; Pulse: 2x monophasic 0.2ms;

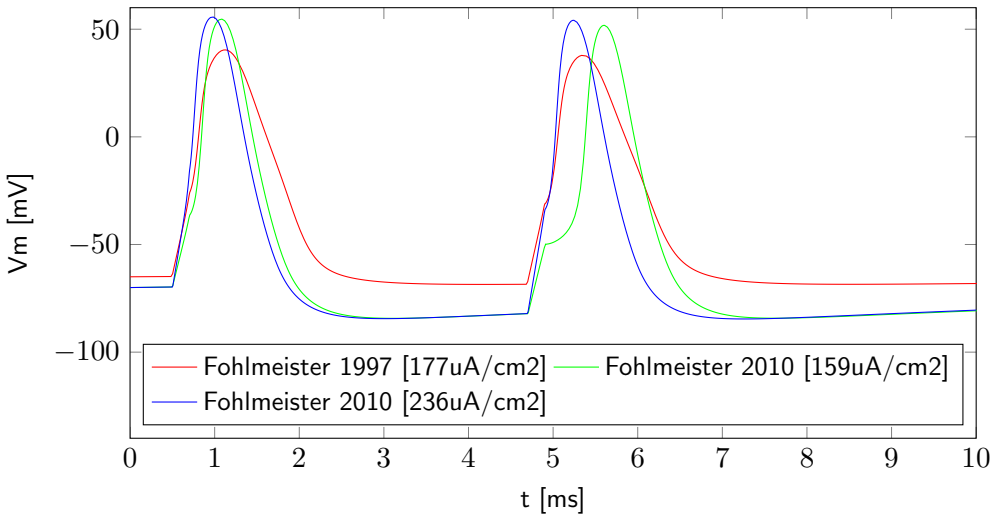


Figure 3.8: Comparison of the inter-peak-time of Fohlmeister et al. 2010 and Fohlmeister et al. 1997 [Fohlmeister et al., 2010, Fohlmeister and Miller, 1997a] with 2 stimuli and an inter-pulse-time t3=4ms. Model properties: Morphology: Single compartment; Channels: Fohlmeister et al. 2010 and Fohlmeister et al. 1997; Temperature: Fohlmeister et al. 1997: 20°C/ Fohlmeister et al. 2010: 23°C; Pulse: 2x monophasic 0.2ms;

Self-Stimulation

At standard configuration, self-stimulation only occurs in Fohlmeister et al. 2010 model. Hodgkin & Huxley and Fohlmeister et al. 1997 do not show any instabilities which compel these models to require higher activation energies. The self-stimulation is only present in single compartment models. In more biologically realistic geometries that include soma, axon and dendrites, the current of the soma gets discharged to other cell parts and therefore stabilizes the electrical system in the neuron. In single compartment, this effect can be reached by increasing the leakage conductance g_L, which leads to an increased i_L. Without these discharging mechanisms and any other input, the Fohlmeister et al. 2010 simulated cat soma spikes after 55ms.

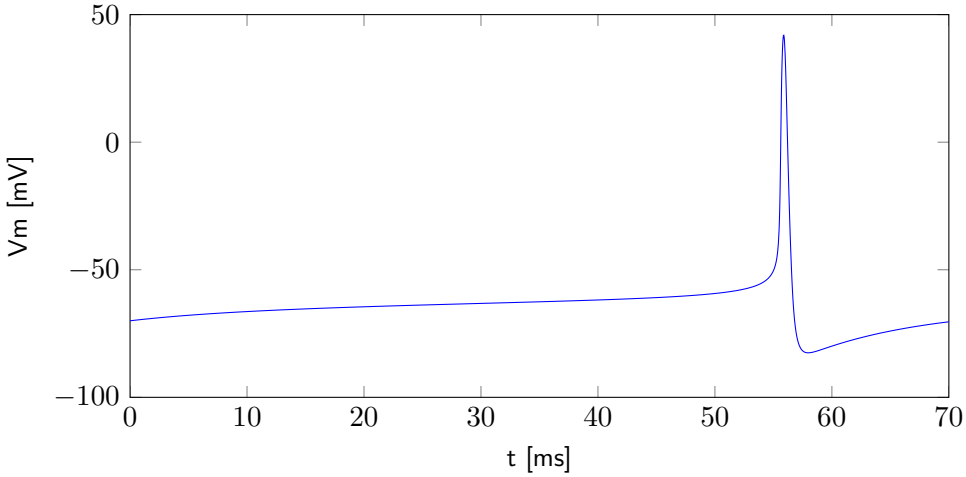


Figure 3.9: Self-stimulation of Fohlmeister 2010, without applying any current. Model properties: Morphology: Single compartment; Channels: Fohlmeister et al. 2010 ($g_L = 0.1 mS/cm^2$); Temperature: 23°C; Pulse: none;

Chapter 4

Electrode Configuration Study

4.1 Methods

All simulations were generated in a 2-step procedure:

Step 1: Finite Element Solution

As the first step, an external potential profile is generated in Comsol Multiphysics using the finite element method (FEM). Hence, a three dimensional model of a small part of a mammalian retina has been generated as a volume conductor, under consideration of various material and geometric properties -as can be found in the literature. This model consists of:

- a block of retinal tissue, size 3x4 cm, formed by 5+2 distinct layers. (Chapter 4.1.1)
- a four-segmented RGC, placed in the retinal tissue to provide exact position for potential measurement, but excluded from FEM calculations (Chapter 4.1.2)
- an electrode carrier consisting of six grounded and one stimulating elements (Chapter 4.1.3)

A tetrahedral mesh was used to compute the electrical potential at each point of the geometry with regards to different electrical properties of the retinal layers. For simplification, this simulation uses a static profile, where

effects like time-dependent change of the potential caused by neuronal activation, electrode impedance, tissue permittivity, and isotropic conductivity are neglected. As border conditions between the electrodes and the nearby tissue, Neumann conditions are used. The presented volume conductor model represents a sufficient method for modelling many simulation tasks.

Step 2: Calculating Membrane Potential

In the second step, the computed external voltages are imported in MATLAB and used to calculate the membrane potentials of the RGC over time. A compartment model was used considering specific ion channel densities for each cell section. The current flow through the cell membrane (Chapter 4.1.2) was calculated following the Fohlmeister et al. 2010 model [Fohlmeister et al., 2010]. The differential equations of the membrane potential were solved by the (non-stiff) ode113 solver included in the MATLAB software bundle.

The procedure to determine activation threshold included increasing the input voltage on the stimulating electrodes, starting from $0mV$ at $-1mV$ increments, until a propagating AP starts to show.

4.1.1 Retinal Tissue Model

The retinal tissue is simulated in a static model by a volume conductor consisting of a passive model of 5 inner layers and 2 border layers (vitreous and choroid). This volume conductor is defined according to physiological criteria. Each layer averages the microscopic tissue properties to a single macroscopic parameter, the specific electric conductivity sigma which is assumed to be isotropic. Real tissue was found to be inhomogeneous and anisotropic, as well as with variations in the thickness of the layers [Karwoski et al., 1985]. Previous simulations showed a time dependence of the potential distribution, which was neglected in this simulation [Dokos et al., 2005].

	thickness [μm]	σ [S/m]	ρ [$\Omega \cdot$ m]
Vitreous layer	25	1	1
RGC layer	20	0.014	70.60
Inner Plexiform layer	35	0.0549	18.2
Outer / Inner nuclear layer	60	0.0167	60
Sub-retinal space (Rods & Cones)	55	0.0787	12.70
Retinal Pigmented epithelium	20	8.13e-4	1230
Choroid [Joarder et al., 2011]	100	0.4348	2.30

Table 4.1: Layers, size, conductivity σ and resistivity ρ of retina model (top-down) [Karwoski et al., 1985]

The electrical and geometric properties are based on experiments in frogs [Karwoski et al., 1985]. On top of the retinal structure, a thin vitreous layer has been placed to achieve a distance of $30\mu m$ between the RGC-layer and the electrodes (Chapter 4.2). The electrode carrier used in clinical applications is an electric insulator such as polyamide foil (used in experiments of Zrenner and coworkers (2010)), and electrically shields the rest of the vitreous body against the retina. Thus, the carrier and the upper vitrous body have no electrical effect and, therefore haven't been modelled. At the bottom of the structure, a choroid layer was added, which delimits the retina and acts as an electrical insulator [Zrenner et al., 2011].

4.1.2 Retinal Ganglion Cell

In the upper area of the RGC layer, a multi-segment neuron was placed, based on a model derived from the tiger salamander RGC [Fohlmeister and Miller, 1997b]. The neuron was inserted into a volume conductor to derive the exact positions where to calculate the membrane potential. This cell was excluded from all FEM solving steps. The neuron was composed of 4 unmyelinated neuronal segments (Table 4.2): the soma was modelled as a sphere with the highest distance to the electrode and partly placed in the inner plexiform layer. Attached to the soma at a 26 degree angle, the axon hillock (AH) is placed. For simplification, the AH is modelled as cylinder and uptakes the entire slope, where morphological studies showed a continuation of the AH aligned to the axon. The 26 degrees were derived by passing the $40\mu m$ segment on a height-difference of $25\mu m$ between sodium channel band (SOCB) and soma. This means that the axon bend is the starting region of the SOCB.

	diameter [μm]	length [μm]	compartments
Soma	16	/	1
axon hillock	1	40	3
SOCB	1	40	3
DA	1	1500	93

Table 4.2: Geometric properties of the multi-compartment RGC model

Because of the electro-physiological and geometric properties of the SOCB, stimulation at electrode locations close to he SOCB results in the lowest threshold. Since the AP is initiated in the axon and in the SOCB first, the active membrane behaviour of dendrites has no impact on the membrane potential during cathodic stimulation from the epi-retinal space. According to the activating function, the passive properties can influence the neuron, but this influence is rather small. As a result of this, the dendrites have been neglected in this simulation to minimize computational effort.

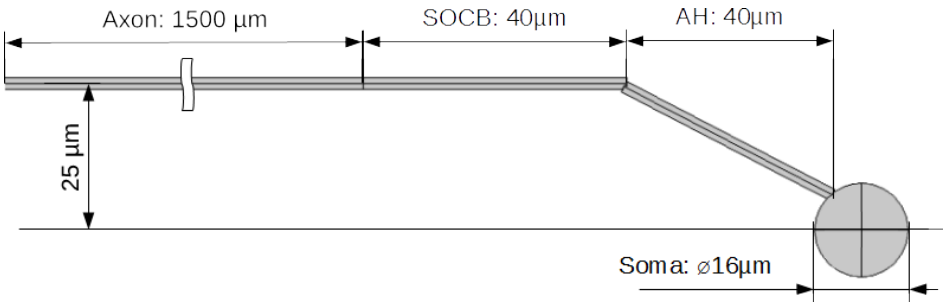


Figure 4.1: Modelled neuronal parts, y-z view

Complementary to the morphological structure, a mathematical model for calculating the membrane potential was added. The neuron was therefore divided into 100 compartments; each between 13.34μm and 16.13μm long, which is coarser than in previous studies where 1μm slices were used [Greenberg et al., 1999]. This decreased precision does not show significantly different results and has subsequently been used for all simulations. The compartments are connected to each other using the electrical network model and the intracellular resistivity was set for all compounds to 1.1Ωm [Coleman and Miller, 1989, Fohlmeister et al., 2010, Rattay, 1999].

The stimulation spot of the RGC was either directly above the sodium channel band (Chapter 4.2.3) or at the center of the axon (Chapter 4.2.2), where the center electrode stayed during distance variations on a constant spot above the neuron.

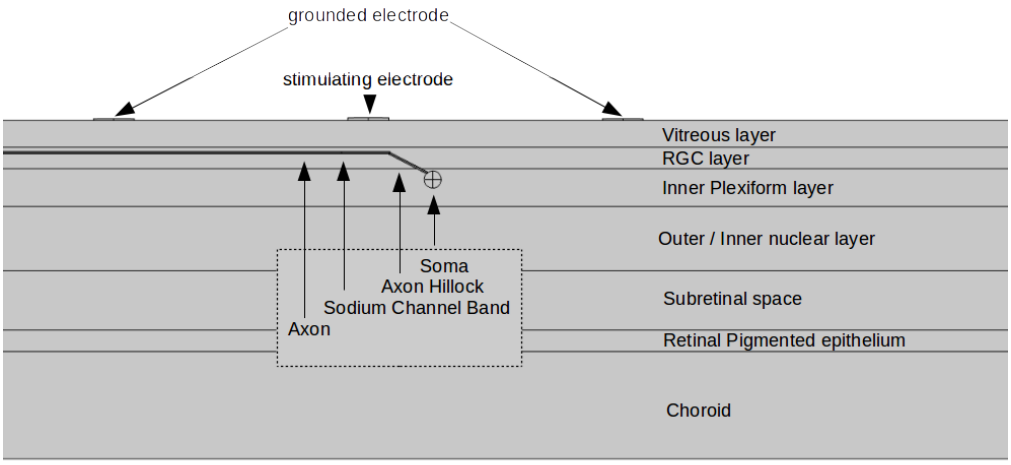


Figure 4.2: Neuron placed in retinal tissue beneath electrode, y-z view

4.1.3 Electrode Model

In order to stimulate the target neuron, $40\mu m$ diameter, disc-shaped platinum electrodes are used. Electrodes of this kind have been used in previous studies [Margalit et al., 2002]. They are modelled as cylinders, embedded in insulating substrate, and have been arranged in a hexagonal configuration in the vitreous body for epi-retinal stimulation. The guard-to-center spacing varies from $3.3\mu m$ to $248.6\mu m$ (minimal distance between center electrode border and guard electrode border, see Table 4.3). Other representations are center-to-center spacing (P), where the distance (y-axis) between the center of the guards is measured. Additionally, at the border-to-border spacing (B) the x-axis the most outer points (left and right) is measured.

For simplification, only one set of electrodes (seven stimulation channels) are simulated. The effect of crosstalk -especially at small electrode spacings- between enabled and disabled stimulating electrodes, can not be simulated on this small setup [Wilke et al., 2011].

	1	2	3	4	5	6	7	8
S	3.3	17.73	46.6	75.47	104.33	133.25	190.94	248.675
P	75	100	150	200	250	300	400	500
B	126	155	213	270	328	386	501	617

Table 4.3: Overview of electrode-distances 1-8 in μm: Space between 2 electrodes (S) (further called guard-to-center spacing), Center-To-Center Spacing (P), border-to-border Spacing (B)

In order to achieve the stimulation of a pattern in the retina, electrodes ideally should inject current in a limited spatial region, which enables focal stimulation of target neurons without interfering with adjoining neurons. To limit the effect of the applied potential on distant parts of the retina, each stimulating electrode is guarded by six grounded electrodes (guards), I - VI in a hexagonal shape (Figure 4.3).

The computation of the finite element model could be massively improved by using symmetry axes. The neuron was only a representation of a single cell in the RGC-layer where its electrical properties are already considered in the macroscopic layer parameters. It therefore, has no influence on the properties of the RGC layer and can be ignored for the finite element solution. Further to this, the geometry could be divided into 6 electrically equal parts, and solving one of these parts is sufficient. This would reduce the computational effort by nearly factor 6. On the other hand, this finite-element model is relatively simple and can be calculated within a decent time.[1]

[1] meshing: 8s, solving 126s

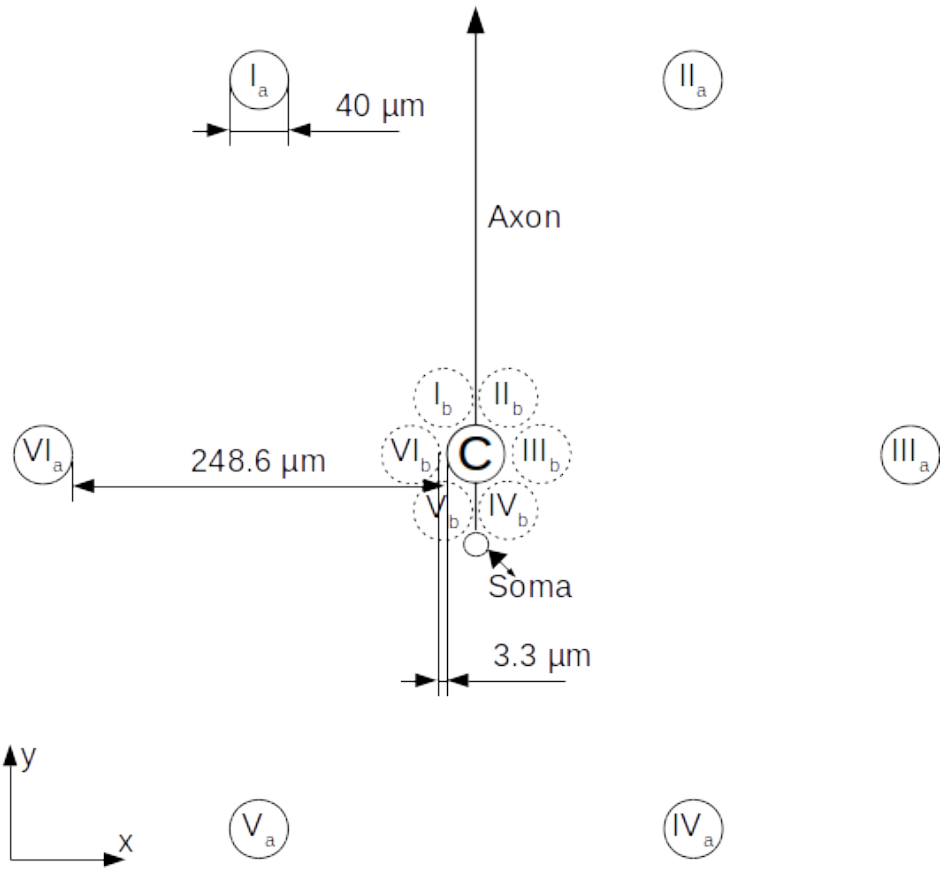


Figure 4.3: Hexagon shaped electrode configuration: electrodes I - VI are grounded (0V) and on the surface of electrode (C) an electric potential is applied. Electrodes with index a show the maximal distance of $248.6\mu m$ to center (guard-to-center spacing), while b indexed electrodes denote electrodes with the minimal distance of $3.3\mu m$ to center (all distances in scale)

4.1.4 Stimulation Pulse

The basic criteria for designing a stimulation pulse for biological purposes, formulated by Merrill et al, reads as follows: *The electrode potential must be kept within a potential window where irreversible Faradaic reactions do not occur at levels that are intolerable to the physiological system or the electrode* [Merrill et al., 2005] Applying this rule in clinical applications is inevitable, but it is still often ignored in simulations. Therefore, the influence of charge balanced pulse compared to monophasic pulse has been analysed in one simulation.

If not explicitly stated otherwise, the simulation was driven by a monophasic pulse profile of $200\mu s$ length (Figure 4.4a), with a negative (cathodic) potential (relative to the grounded return electrode) applied on the center electrode. In general, negative pulses have lower thresholds than positive ones [Dokos et al., 2005, Rattay and Resatz, 2004]. Without any stimulating voltage, the Fohlmeister et al. 2010 neuron spikes after 14 ms, independent of the guard-to-center spacings [Fohlmeister et al., 2010]. All simulations were running for $5ms$, and an AP was taken into consideration only if the AP occurred and reached its maximum within this time span. Because of these short simulation times, the previously described self-stimulation does not significantly influence the results.

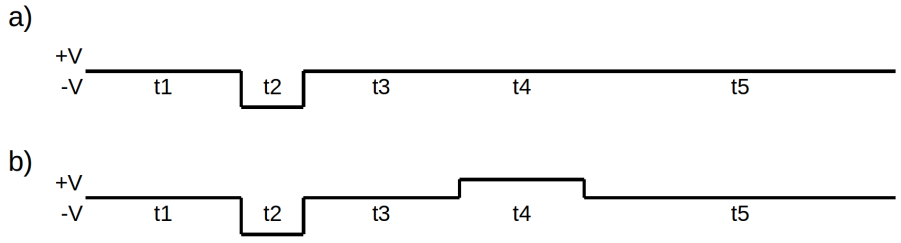


Figure 4.4: Function of input potential over time. t1 to t4 are in scale. t1=0.5ms, t2=0.2ms, t3=0.5, t4=0.2ms, t5=3.5ms (not in scale) **a)** monophasic pulse **b)** Cathodic-Anodic pulse: charge balanced input pulse

Destructive stimulation effects can be caused by two reasons: one of which is exceeding the charge and current density limits, the other being charge accumulation. While current and charge density will be topics in the following pages, an adaptation of the stimulation pulse can avoid charge accumulation. An additional discharging pulse needs to be added in order to achieve a charge balanced stimulation. This pulse will prevent net charge accumulation by unloading the capacitive properties of the tissue

in the chemical surrounding of the electrode. A disadvantage of this discharging pulse is its stimulation-inhibiting effect. To weaken this effect, the inter-pulse intervall (t3) is used to prevent the AP suppression. If ignored, destructive effects like electrolysis, harm the tissue and also the implanted electrodes. Furthermore, dissolving of the electrode can occur whereby some metal ions have a toxic effect on neural tissue.

Since previous simulations have shown a small threshold-decreasing effect of the second pulse in Fohlmeister et al. 1997 model, the influence was found nearly negligible [Fohlmeister and Miller, 1997b, Jeng et al., 2011]. The response of a biphasic pulse on Fohlmeister 2010 has not yet been documented in literature. In previous experiments on feline retinas, cathodic-first stimulation has been found to be more effective than anodic-first stimulation [Dokos et al., 2005, Schanze et al., 2002]. Therefore, the voltages carry a negative sign.

The second pulse however, can also elicit an undesired AP. Due to this, a bipolar pulse- where the second pulse had double the pulse length, but half of the pulse size (and therefore an equal charge) (Figure 4.4b), separated by $0.5ms$ inter-pulse separation time [Jensen and Rizzo, 2008]- has been used as stimulation pattern on the SOCB region.

Interestingly enough, the activation threshold increased an average of 18% in comparison to the monophasic pulse, but the sensitivity towards sharpness of the voltage profile decreased. Figure 4.16 shows the effect of a biphasic pulse on the minimal stimulation potential (red dashed line), in comparison to a monophasic pulse (blue solid line).

4.1.5 Finite Element Method (FEM)

Errors and Validation

The finite element method is a powerful tool for analysing complex physical problems. For laymen especially, it is tempting to trust results based on predefined settings and automatic configuration, since it can lead to fast results and impressive images. These simulations might, however, miss the actual point of acquiring realistic results and become worthless. Therefore Verification and Validation (V&V) is an important part of finite element analysis (FEA) [Pointer, 2004].

In the present simulation, the numerical error was reduced from several thousand percent at automatic settings and simple geometry, to less than 1%, by a bundle of optimizations. All numbers are generated by SOCB

stimulation, center-to center spacing: $190\mu m$ and 1V input potential. Furthermore, it has to be kept in mind that an error which is caused by insufficient geometry will propagate, and will amplify any future errors that occur further on in the process.

In general, the errors in FEA can be divided in 3 categories [Shah, 2002]:

- **User Error**
  Error caused by incorrect use of the FEA-software.
- **Modeling Error**
  Error caused by incorrect or insufficient geometry, boundary conditions, solver settings and physical model.
- **Discretization Error**
  Error caused by meshing method.

It is difficult to verify the accuracy of the solution, especially if there is no experimental data to compare with. There is no exact procedure to measure the accuracy of the FEA solution, since different physics require different methods. Two distinctive tools for investigating the error rate are used for the present simulation, and both criteria must be satisfied:

1. **Error Norm**
   The model has to follow the basic law of energy conservation. This means that for a stationary (time-independent) solution, all currents entering the model will need to exit as well. To measure the error norm of the entire model, the percentage of incoming and outgoing current has been calculated. As a rule of thumb, these values have to be below 15% for the entire model, and below 10% for the region of interest [Shah, 2002]. If not explicitly mentioned otherwise, all numerical error rates in this simulation were calculated by this method and improvements were regarding the final solution.
2. **Convergence Plot**
   The convergence plot (Figure 4.5) is one of the most important indicators of an accurate solution. It describes the relative difference between the last two iteration steps. This curve needs to be verified graphically and must provide at least 3 iterations. Moreover, the error rate has to be continuously decreasing. Still, even with a promising convergence curve, the solution might contain significant errors [Pointer, 2004].

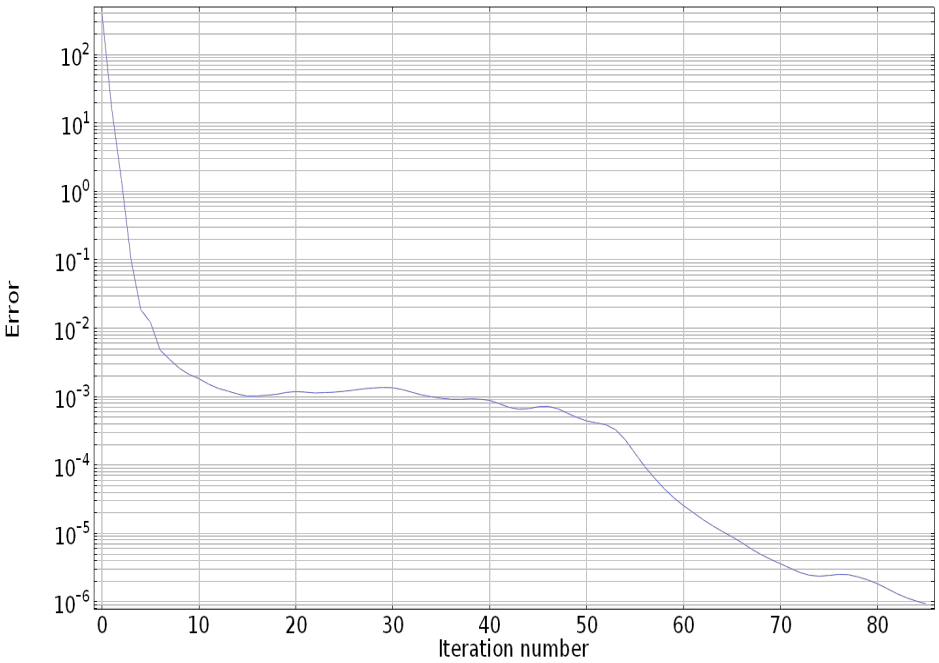


Figure 4.5: Convergence plot of SOCB stimulation, guard-to-center spacing: $190\mu m$. Please note: y-axis is logaritmic

Solver Optimization

The solver tolerance is the allowed difference (in magnitude) between two iterations in the final solution. If this value is reached, the solver accepts the solution and does not run further iterations. This tolerance has been reduced from 10^{-4} (default settings) to 10^{-6}. The default values are hardly applicable to electrical simulations and can usually be reduced with up to 10^{-8}. Lower values would increase the error rate again, because of the numerical limitation in double precision floats in the underlying computer hardware. The current flowing through the isolated BC ($30 \cdot 10^{-12} A$) might also be caused by this numerical limitation. This solver customization decreased the numerical error rate from 31.88% to 0.03%, but increased the number of iterations from 15 to 64, as well as doubling the calculation time.

Geometrical Adaptation

FEM models can be imported from CAD software, and these models are to a great extent detailed. And so, they often contain structures which have no influence on the physical behaviour in the region of interest. Still, these structures need to be meshed and calculated. This leads to a highly increasing number of mesh elements and various other problems, further on. The convergence analysis, where the numerical difference between each mesh refining step is compared, is then hardly possible. Therefore, an accurate geometry is only needed in critical sections of the model [Kurowski, 1994]. For the model in the current simulation, this rule implies that the RGC has been excluded from meshing and all further calculations, because the diameter of $1\mu m$ is much less than even the smallest mesh element size ($6\mu m$).

In constructional engineering as well as in FEA, sharp corners should be avoided. Meshing these corners can lead to singularities where with decreasing element size, the local error in the mesh-element increases. The FEA tools usually have built-in methods to deal with these situations and try to minimize the global error in the entire model, but the singularity must not be in the point of interest. Avoiding these singularities however, improves the accuracy and decreases the number of mesh elements. Here, cylindrical disc electrodes were used instead of cornered cubic electrodes. This replacement increased the precision by reducing the amount of mesh elements.

Another step taken to reduce the number of mesh elements was removing electrically irrelevant parts. Therefore, the vitreous layer above the element-carrying foil has been removed, because this tissue cannot receive any current through the electrically insulated foil.

All these steps of geometrical optimization combined, were able to reduce the numerical inaccuracy by over $40,000\%$.

Physics

A foible of Comsol Multiphysics is that it has a higher precision of 'Terminal' node boundary conditions than 'Electrical Potential' boundary conditions, on coarse and standard mesh. For infinite fine mesh, the results would be equal[2]. Since the individual mesh for the regions of interest is very fine, the final result could not be improved by switching from 'Electrical Potential' to 'Terminal'. For other problems however, this information might be useful.

Meshing

Meshing can be considered a sampling method for the model. It divides the geometry into small pieces for subsequent use in solving the physical equation, by the solver. This way, the physical equations are evaluated individually for each of these elements. These elements are used for presenting the solution as well. Theoretically, in a perfect mesh, adjacent mesh elements would have continuous solution. In practice, there is no 'ideal' mesh for the entire model. A too-fine mesh acquires too much computational effort, while a too-coarse mesh might not be able to simulate the physical questions properly. Therefore, the mesh should be very fine in the region where the physical activity is taking place, and in the regions of interest and coarse in other regions. For distinguishing where these places of interests are, the experience and intuition of the engineer is needed. Solving the entire model with a coarse mesh can help find these regions of interest, but should not be trusted unquestionably. A too-coarse mesh might miss important regions if, for instance, the current source and drain is inside one single mesh element.

As a rule of thumb, each 90°arch should have at least eight meshing elements. Therefore, the amount of mesh elements along each electrode edge has been increased to a fixed number of 50 elements, because this is the region where the highest currents are expected.

For the present simulation, the adopted values, compared to an automatic generated mesh, can be found in Table 4.4. As meshing element, free tetrahedra is used (second-order element).

[2]http://www.comsol.com/community/forums/general/thread/32552/

	auto mesh	electrodes	RGCl/vl	rest
max. element size [μm]	220	140	140	320
min. element size [μm]	16	6	6	40
max. element growth rate	1.4	1.35	1.35	1.45
resolution over curvature	0.4	0.3	0.3	0.5
resolution over narrow regions	0.7	0.85	0.85	0.6
number of mesh elements	499157	813090		
solution time [s]	63	126		
numerical accuracy	98.3%	99.97%		

Table 4.4: Mesh configuration compared: automatic generated mesh, size 'Fine' for the entire model, compared with adopted mesh for different regions of the model. The Retinal Ganglion Cell layer (RGCl) and the vitreous layer (vl) are meshed with the same properties (RGCl/vl).

4.2 Electrode Distance Optimization

The main goal of electrode spacing optimization is to determine the configuration containing the highest amount of electrodes on the implant, considering counteractive electrical effects.

Complex vision requires several hundreds or thousand electrodes [Margalit et al., 2002, Wilke et al., 2011]. Despite some research which suggests increasing the resolution without increasing the number of electrode channels by pulse modulation, like from Behrend and coworkers (2011), the number of electrodes on a prosthesis remains a key limitation [Behrend et al., 2011].

To place the maximum amount of electrodes on the limited size of retinal tissue, minimal distances between electrodes are aspired to, while respecting the limits of charge density and keeping an eye on selectivity. This simulation demonstrates the relation between center-to-guard spacing and energy consumption. Table 4.5 shows the theoretical maximum amount of electrodes for each configuration on a human retina. Considering an average diameter of the human retina of 22mm and 72% of the interior globe taken up by the retina, the human retina has a size of $1094mm^2$ [Kolb et al., 1995], compared to the maximum which could be deployed on the $9.3mm$ implant used by used by Zrenner and co-workers (Supplementary) (2010) [Zrenner et al., 2011].

(a) center-guard spacing $[\mu m]$	**(b) configurations area area** $[mm^2]$	**(c) configurations per retina**	**(d) configurations per implant**
3.30	0.01	99,817	849
17.73	0.02	56,153	477
46.60	0.04	24,954	212
75.47	0.08	14,036	119
104.33	0.12	8,983	76
133.25	0.18	6,234	53
190.94	0.31	3,509	30
248.68	0.49	2,245	19

Table 4.5: Maximal number of electrodes on a retina with a space of $1094mm^2$ [Kolb et al., 1995] (a) guard-to-center spacing of the electrode configuration; (b) the area, taken up by one configuration, including a gap to neighboring configurations; (c) amount of electrode configurations which can be deployed (d) amount of configurations which can be deployed on a 3 x 3.1 mm implant, such as used by Zrenner and co-workers (Supplementary) (2010) [Zrenner et al., 2011];

4.2.1 Potential Shape on Various Spacings

To visualize the effects of the electric field on different guard-to-center distances, the electrode configuration (Figure 4.3) has been placed on top of the volume conductor, which leads to a distance of $55\mu m$ from the RGC layer. A potential of 1V has been applied and the potential reaching the RGC at the steady state has been captured. The distribution is linear to the applied voltage, therefore the potential is denoted in percent.

Small spacing

The electrodes generate an electric field, which arouses an electric current. On very small guard-to-center pixel spacings, these high currents occur on the same plane between the stimulating, and the guarded electrodes and the penetration depth is comparably small. A higher stimulation potential for generating a stronger field (therefore higher currents) is required to overcome this problem, which has a lot of disadvantages like; tissue heating, increased chance of crosstalk, chance of eliciting an AP in underlying dendritic tissue, increased size of electrodes, and possible harmful effects to the underlying tissue over a specific threshold [Greenberg et al., 1999].

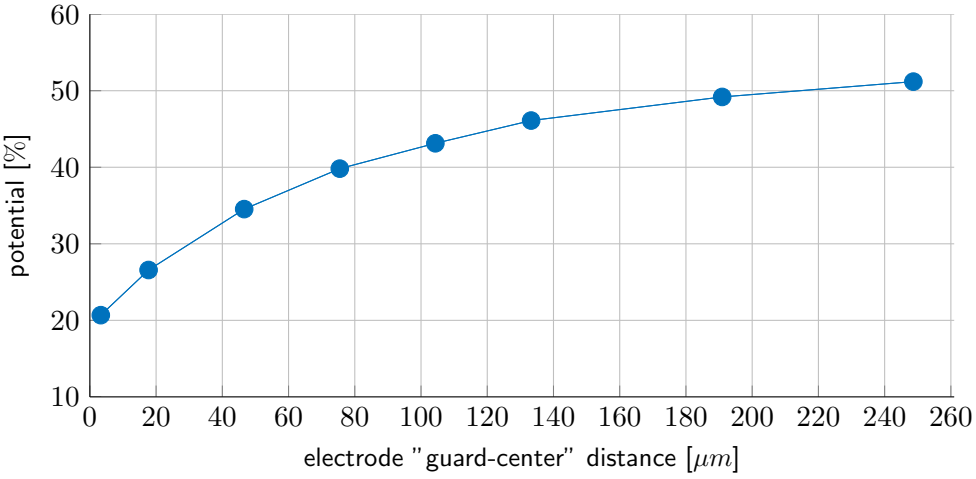


Figure 4.6: Relative potential arriving at RGC layer, guard-to-center spacing varying

Large spacing

A larger distance between the guards and the center lead to an increased spatial volume of stimulation, while injecting a minimum of current, but it also reduced the possible resolution of the implant, because of the limited size of the retina. Additionally, the potential gradients in large spacings are lower. This means that the potential difference between the neuron and the surrounding extracellular medium is less than in smaller spacings. This causes an increasing current demand for eliciting an AP.

Potential contours

Responsible for variations in the potential, are different conductivities among the retinal layers. A higher electric resistance in a layer leads to a stronger curvature of the electric field, and further, to the extracellular potential profile. Also, sharper contours cause higher values in the activating function [Resatz, 2005].

Figures 4.7 - 4.10 picture the focal potential gradient of the electric field beneath the stimulating electrode, including the increasing penetration depth, and widening of the potential contours. For demonstrative and comparative reasons, a stimulating voltage of +1 V has been applied to all configurations. These values are comparable to medical application: In the neuronal prosthesis by Zrenner and co-workers (2010) a voltage between 0.5V and 2.3V was applied [Zrenner et al., 2011](Supplementary).

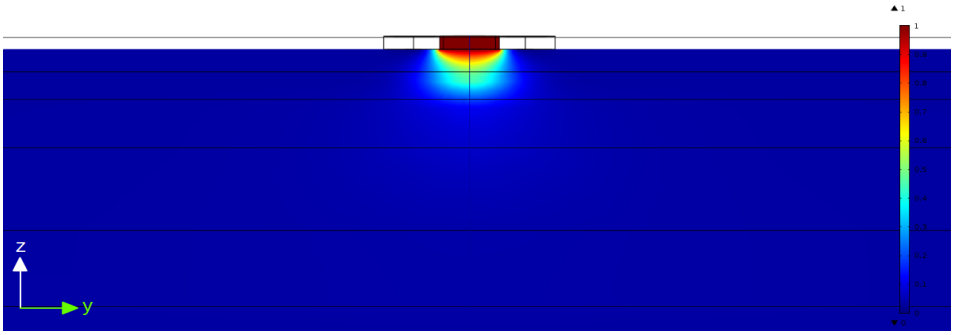


Figure 4.7: electric potential through retinal tissue, guard-to-center spacing: $3.3\mu m$, center stimulus: 1V (red), guards grounded

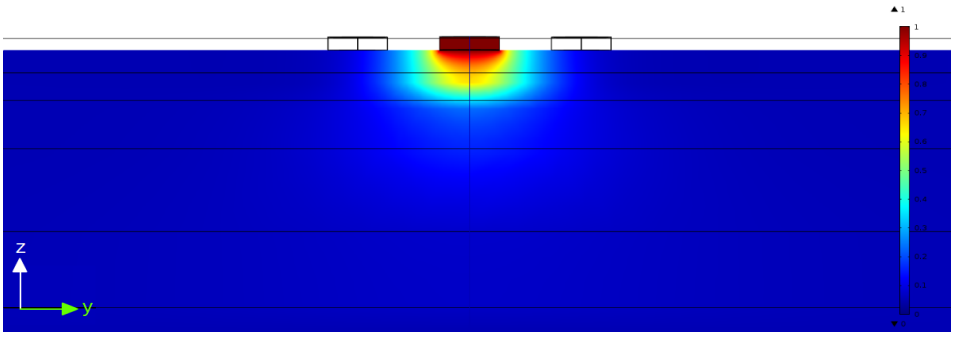


Figure 4.8: electric potential through retinal tissue, guard-to-center spacing: 46.6μm, center stimulus: 1V (red), guards grounded

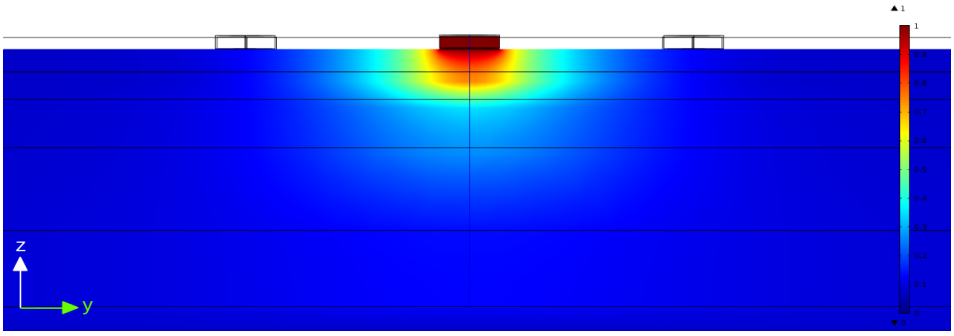


Figure 4.9: electric potential through retinal tissue, guard-to-center spacing: 133.25μm, center stimulus: 1V (red), guards grounded

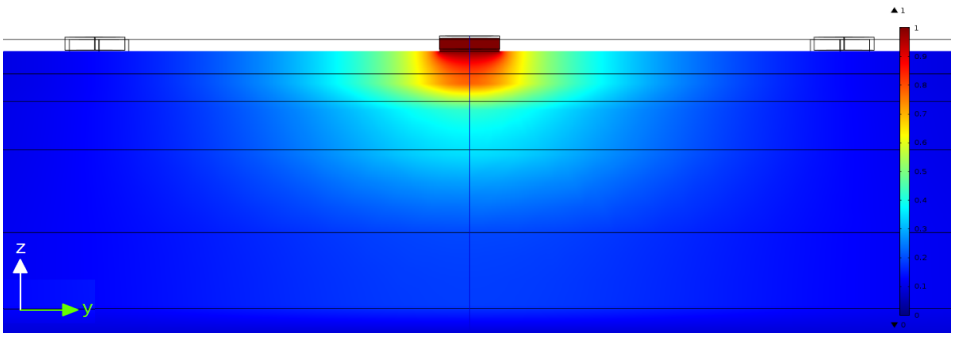


Figure 4.10: electric potential through retinal tissue, guard-to-center spacing: 248.675μm, center stimulus: 1V (red), guards grounded

4.2.2 Axon Stimulation

Given the fact that the effects limiting the spatial excitation have been previously described, the following simulation shows the optimized guard-to-center electrode spacing for low-sensitivity distal axon of the RGC. This means the combination of the soma, SOCB and axon was simulated according to Figure 4.1 and 4.2, but the center electrode of the stimulation was above the distant axon, 403μm away from the special sodium channel band.

To find the optimal distance of the guards to the center, the distance has been varied and the minimal electrode voltage for starting an AP has been measured. The vertical distance between the neuron and the electrodes was kept constant at 30μm to the axon and 55μm to the soma.

This minimum was found at a guard-to-center spacing of 46μm, which results in the ideal guard-to-center spacing for axonal stimulation. This means that a voltage of $-65mV$ is sufficient to start an AP in the 46μm configuration, while the required electrode voltage for stimulation in different spacings is higher. The least efficient point has been reached at 248μm, where $-86mV$ is required to start an AP. However, the electrode configuration can easily increase the guard-to-center distance and even become smaller than 3.3μm (Figure 4.3). These are not the limits of the highest threshold, but of the lowest.

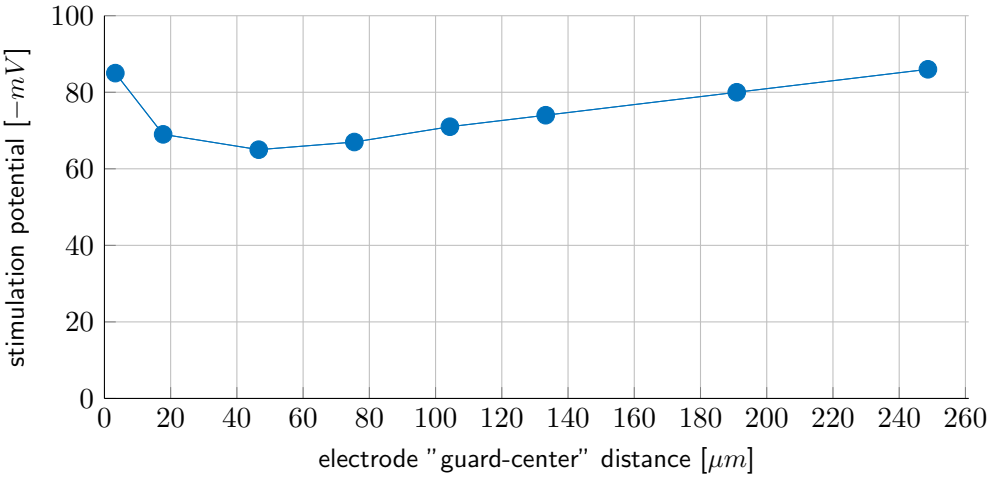


Figure 4.11: Minimal electrode voltage for eliciting an AP on varying guard-to-center spacings; computational FEM error rate (max): 0.186%

Action Potential

Figure 4.12 illustrates the corresponding AP on various guard-to-center spacings, measured on the compartment directly underneath the center-electrode (compartment 56). The stimulation potential was set to $-86mV$ for all electrode spacings, which is sufficient to start an AP at highest thresholds. This is important to ensure a stimulation at the same time and remain comparable. In comparison, Figure 4.13 shows the activating functions for all guard-to-center sizes of this simulation. Clearly visible is that the activating function has higher values at small guard-to-center sizes, despite the smaller penetration depth.

The time between the stimulus and the peak of the AP differs - configurations with lower threshold excite earlier than those with higher thresholds, while lower threshold configurations on the other hand, spike slightly higher ($37.90mV$ for $248\mu m$ compared to $36.02mV$ for $46\mu m$ spacing). Clearly visible is the stimulation time of (0.5 ms - 0.7 ms). In the configurations where the stimulation potential was closer to the lower threshold, the AP did not spike immediately. However, the different AP shapes seem to be caused by a variation in the potential difference between threshold and stimulation potential.

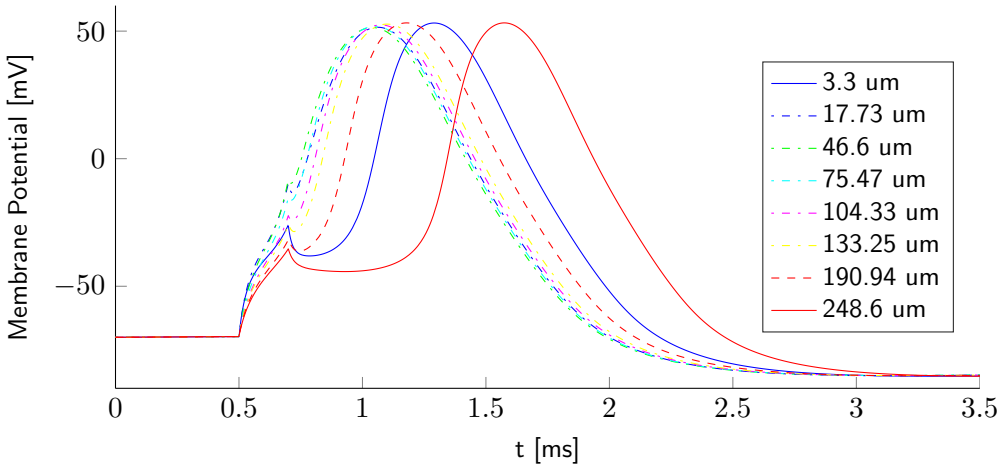


Figure 4.12: AP elicited by $-86mV$ for each guard-to-center spacings, DA stimulation

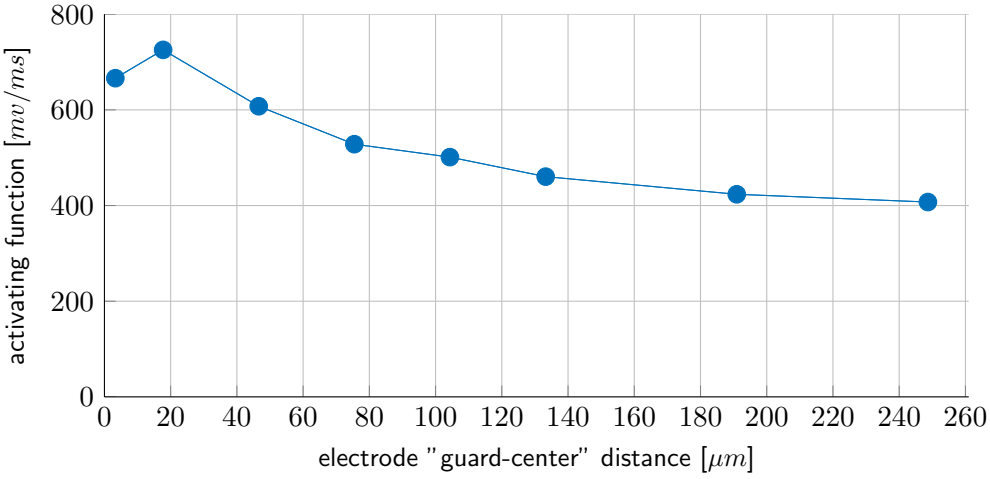


Figure 4.13: Activating function for stimulation potential of $-86mV$ for each guard-to-center spacings

Safe Charge Density

Previous experiments showed a decreasing threshold current with decreasing electrode diameter [Sekirnjak et al., 2006]. Since the voltage and the current form a linearly connected system with only ohmic resistance (like the retinal tissue), the key feature for limiting small disc sizes is the current density. Current density and charge density are linked: the charge density is derived from a uniform current density distribution over the electrode surface [Greenberg et al., 1999]. Therefore, the charge density and the current density graph show an equal shape (Figure 4.15 and 4.17).

Exceeding the safe charge density limits is one of the destructive effects which has to be avoided. If not, the high charge might destroy the tissue, as well as the electrode. To stimulate the distant axon, the charge density of $960\mu C/cm^2$ for the smallest configuration exceeds the $800\mu C/cm^2$ limit (Figure 4.15), while including the SOCB does not exceed the critical limit (Figure 4.17). $800\mu C/cm^2$ is assumed as the lowest threshold for safe stimulation in human and rabbit retinal tissue, using platinum electrodes [Margalit et al., 2002, Weiland et al., 1999]. In physical applications, the charge density is not equally distributed over the electrode (like in this simulation), but slightly concentrated in the edges, which would need even lower safe charge limits. Clinical applications tried therefore, to stay significantly below this threshold, like $600\mu C/cm^2$ for platinum electrodes [Zrenner et al., 2011].

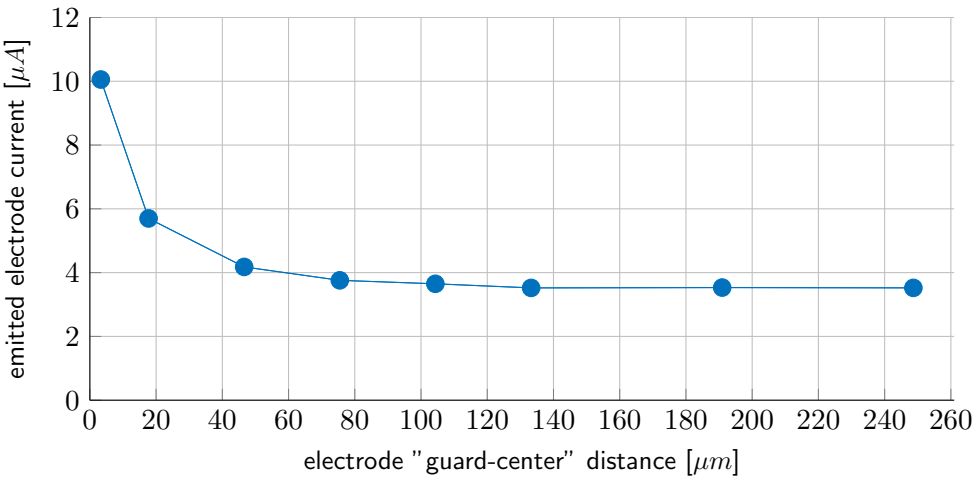


Figure 4.14: Threshold current for the center electrode at distant axon stimulation

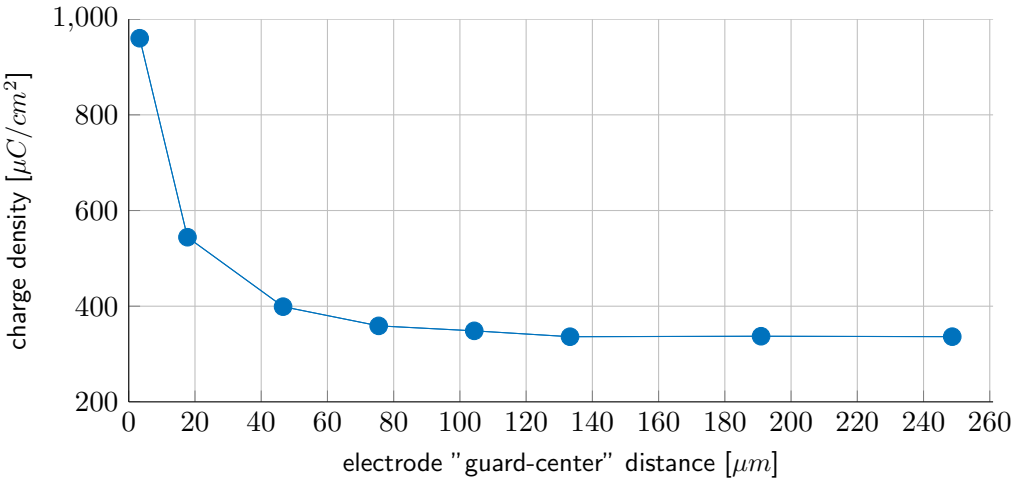


Figure 4.15: Charge densities for threshold stimulation of distant axon stimulation at each electrode spacing

A common approach to increase the electrode area is to roughen the electrode surface, obtaining a granular surface. Especially in platinum, this effect can significantly increase the electrode area (up to factor 100) [Mathieson et al., 2004, Oh et al., 1996].

4.2.3 Sodium Channel Band Stimulation

The SOCB is a short (about 40 μm in length), highly sensitive segment of RGCs, with an increased number of voltage gated sodium channels. This high channel density occurs adjoining to the axon hillock, around $40\mu m$ in distance from the soma [Fried et al., 2009]. The length, position and the conductance of the SOCB differ on various cells, as does the threshold. In this section, the sodium conductance $\bar{g}_{Na}$ is assumed to be up to 40 times higher than in the distal axon. In RGC, the SOCB is assumed to be in a range up to $2800 mS/cm^2$, compared to $124 mS/cm^2$ in axon [Jeng et al., 2011, Jensen et al., 2003]. For the present simulation, the channel density was set to $1240 mS/cm^2$, 10-fold higher than the channel density of the distal axon and within the range of previous work, which is sufficient to see the SOCB behaviour of recent studies in the simulation.

Minimal Electrode Spacings

The minimal electrode voltage for a successful stimulation is $-10.5mV$ for a guard-to-center spacing of $104.33\mu m$ using a monophasic pulse, and $-14mV$ for $75.47\mu m$ using a biphasic pulse, respectively. The low stimulating threshold remains nearly constant on higher guard-to-center spacings, which does not allow specifying an optimal distance, but a worse spacing which is $3.3\mu m$. Considering the goal of a small radius, the best distance is $104.33\mu m$ using a monophasic pulse, and $75.47\mu m$ using a biphasic pulse. Furthermore, these results suggest that the sharp potential contours only play a minor part in the SOCB stimulation, and no role at all in a biphasic pulse stimulation.

Previous studies found the excitability of the SOCB to be two to five times higher than in distal axon segments, which is similar to the factor found in this simulation [Jeng et al., 2011]. Comparing Figure 4.11 and Figure 4.16, both at a guard-to-center spacing of $46\mu m$, the axonal stimulation requires 4.5 times higher stimulation voltage than the SOCB stimulation ($-63mV$ vs. $-14mV$).

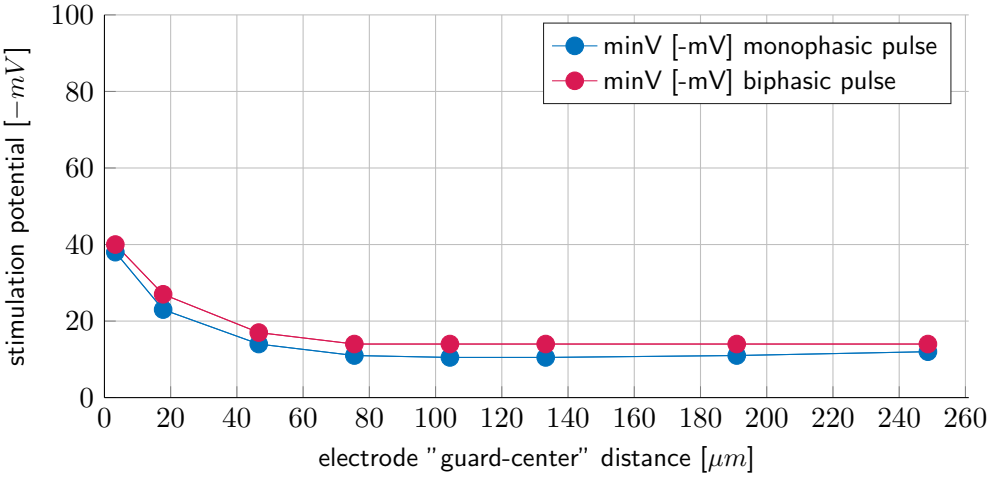


Figure 4.16: Minimum electrode voltage for eliciting an AP on varying guard-to-center spacings; computational FEM error rate (max): 0.179%

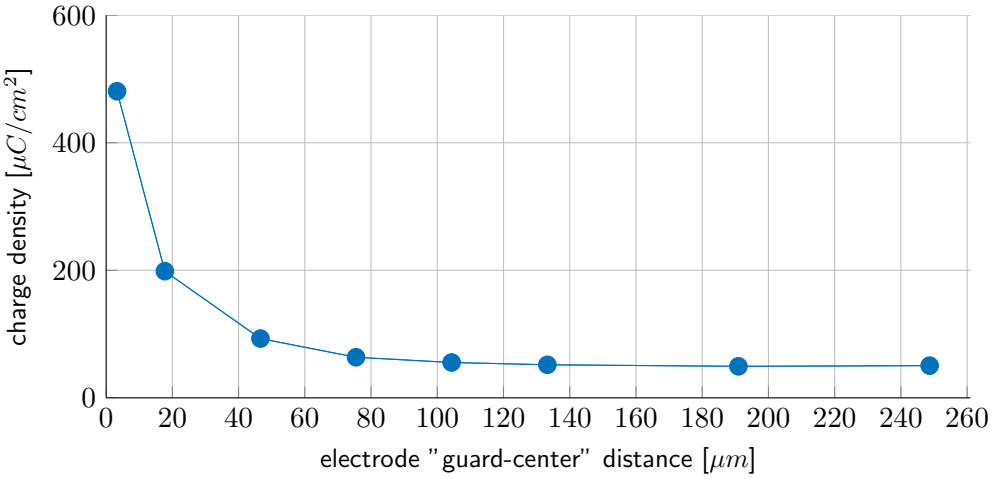


Figure 4.17: Charge densities for the lowest threshold stimulation at each electrode spacing: none of the shown electrode spacings reaches the safe charge density threshold of $800\mu C/cm^2$ at SOCB stimulation [Margalit et al., 2002]

Action Potential

The SOCB has the lowest threshold among the entire neuron, and is also responsible for the lowest stimulation thresholds closest to the stimulating electrode [Fried et al., 2009]. The stimulating potential therefore, was adopted for measuring the influence of the guard-to-center spacings directly above the SOCB, from $-80mV$ in previous simulations to $-40mV$. An equal stimulus was applied for all geometries, and the responding AP measured. However, the action potential does not show any significant change in shape or duration on different guard-to-center spacings (Figure 4.18). As an exception, both of the smallest electrode configurations need longer to arise an AP, which can be explained by their higher demand on stimulation energy. Therefore, the $-40mV$ is close to the minimum threshold for the smallest configuration. Surprising is the difference to Figure 4.12, where the axon does show different AP behaviour on variation of the electrode spacing.

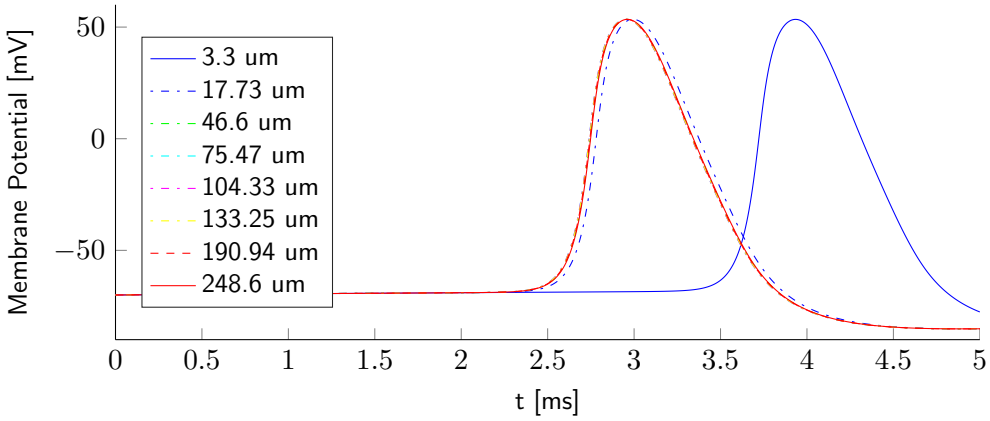


Figure 4.18: AP elicited from a $-40mV$ stimulation potential for each guard-to-center spacings, SOCB stimulation. The AP for guard-to-center spacings of $46.6\mu m$, $75.47\mu m$, $104.33\mu m$, $133.25\mu m$, $190.94\mu m$ and $248.6\mu m$ overlap entirely.

Ion Channel Gating

As previously described, different guard-to-center spacings require different stimulation voltages, which raises the question whether this different voltage (and subsequently the different current) leads to any changes in the behaviour of the ion channels. Therefore, three gating kinetics of Fohlmeister, Cohen and Miller 2010 have been compared with each other. These are kinetics of the smallest ($3.3\mu m$), the kinetics of the electrode configuration which first reached minimum threshold ($46.6\mu m$) and the kinetics of the configuration with the highest guard-to-center spacing. For stimulation, the threshold electrode voltages of $-12mV$, $-14mV$ and $-39mV$, respectively, were applied.

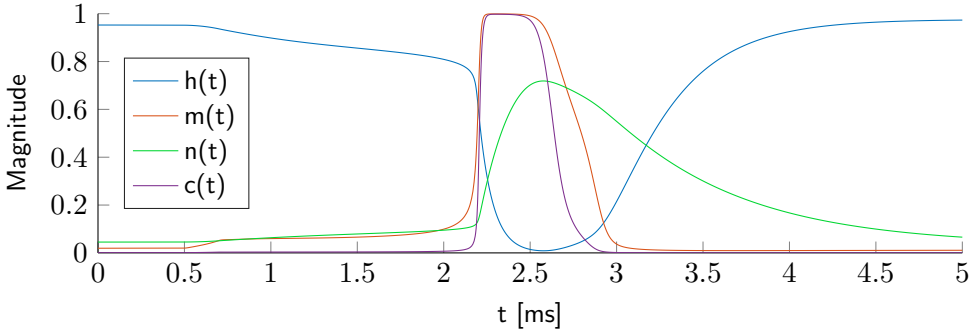
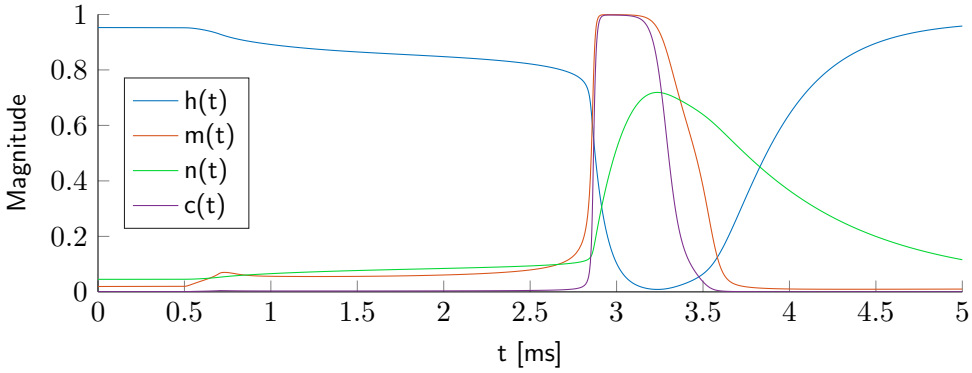


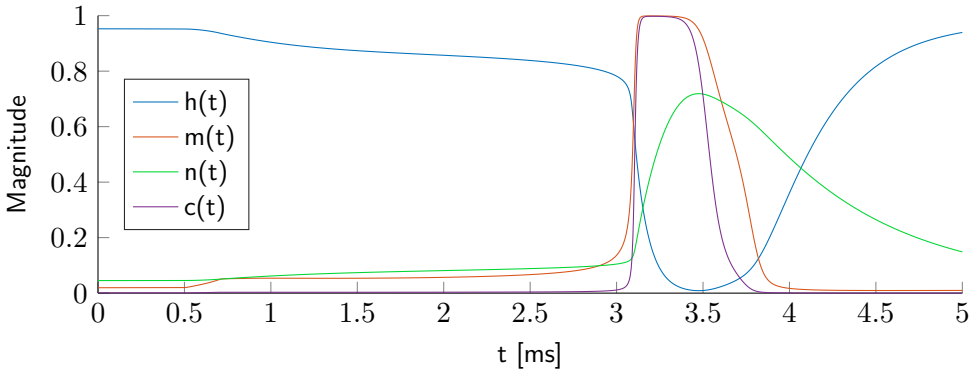
Figure 4.19: Gating dynamics of SOCB stimulation $-15mV$ ($1mV$ above threshold), $46.6\mu m$ guard-to-center spacing

The first notable difference in Figure 4.20 is the fastest reaction time during stimulation at the electrode geometries with higher guard-to-center spacing. This can be explained by the resolution using $1mV$ steps, in order to find the threshold voltage. An increase of $1mV$ can cause a $0.9ms$ faster elicitation (measured at the point where sodium channels reach 50%), as illustrated in Figure 4.19: the effect of a $1mV$ increased input voltage on $46.6\mu m$ guard-to-center spacing (Figure4.20 (b))

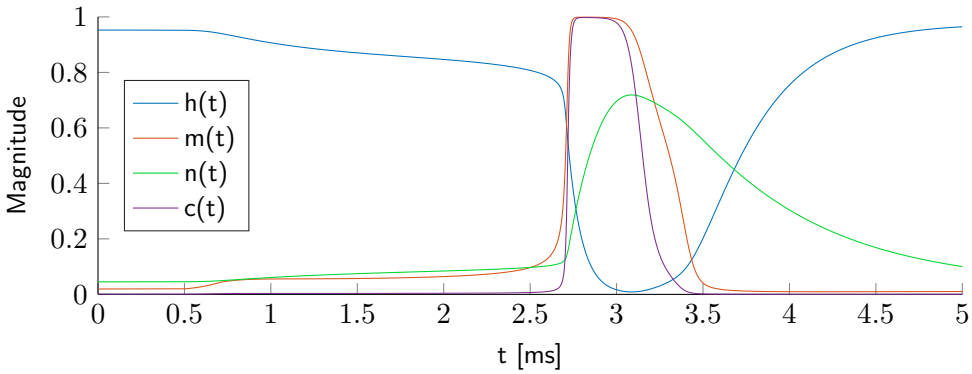
Furthermore, a small spike of the sodium channels $m(t)$ at $0.7ms$ occurs in the smallest geometry Figure 4.20 a), which exceeds the potassium channels $n(t)$ for a short timespan. This is attended by a partially closing inactivating sodium channel $h(t)$. As Figure 4.20 c) illustrates, this sodium channel spike is not correlated with the faster elicitation. The timespan in which the channels are open seems to be independent of the electrode spacing: $0.34ms$ the calcium channel, $0.44ms$ the sodium channel, and the potassium channel $0.65ms$(measured at 50% of gating activity).



(a)



(b)



(c)

Figure 4.20: Gating dynamics of an SOCB compartment: (a) SOCB stimulation $-39mV$ (lower threshold), $3.3\mu m$ guard-to-center spacing; (b) SOCB stimulation $-14mV$ (lower threshold), $46.6\mu m$ guard-to-center spacing; (c) SOCB stimulation $-12mV$ (lower threshold), $248\mu m$ guard-to-center spacing;

4.3 Selective Stimulation

To generate complex images, it is necessary to excite specific neurons exclusively by stimulation, which is generally understood as selectivity. If only specific neurons are stimulated, the brain is able to interpret these stimulations as a pattern and consequently generate the desired image. The unsolicited co-stimulation of adjoining neurons can occur either by stimulating bypassing RGC, or by stimulating bipolar cells which are placed underneath the superficial RGC (z-axis). A possible way to reduce this co-stimulation and increase selectivity would be to stimulate the SOCB just above the threshold potential. This would cause the generation of AP in a single region, while passing axons originating from distant RGC remain unaffected. Neurons underneath the SOCB would obtain an insufficient potential to excite. A different approach would be the adoption of the pulse length and form, to achieve selectivity.

There have been between 13 and 17 different RGC types discovered in mammalian retinas [Fohlmeister et al., 2010], however it remains unclear if all types have an SOCB. Among the cells where an SOCB have been found, the sodium channel density is varying in different RGC types. The length and channel density of SOCB has also been found to be a governing factor of activation threshold [Jeng et al., 2011]. This suggests different stimulation thresholds among these RGC, which is an additional hurdle in stimulating single cells. While the threshold difference between SOCB and distal axon stimulation was according to previous simulations of factor 2 to 3 (Chapter 4.2, Figure 4.11 and 4.16), this factor seems to depend on the guard-to-center spacing as well.

In epi-retinal experiments of Sekirnjal and coworkers (2006) on mammalian retinas, a link between electrode size and selective stimulation has been found [Sekirnjak et al., 2006]. Electrodes with a diameter of $10-15\mu m$ are suggested to be the optimal size for selective single cell stimulation, because of the high spatial resolution. By varying this parameter, an undesired stimulation of underlying neurons can be limited.

Despite all these difficulties, it seems to be possible-by choosing an appropriate electrode size spacing-to increase the selectivity to stimulate single neurons, or tiny groups of neurons [Resatz, 2005, Sekirnjak et al., 2006].

4.3.1 Activation Map

Causes of Co-Stimulation

As previously mentioned, the neuronal part closest to the stimulating electrode is not necessarily the part with the lowest stimulation threshold. The most important reason for this is based on the activating function which says that the key factor for elicitability is foremost, a keen potential difference of the electric field. The absolute intensity of this field is a minor point. If the potential difference of the external voltage between 2 compartments is too small, the occurring currents between the compartments are not sufficient for striking an AP [Resatz, 2005].

A second effect which causes the stimulation of a different neuron than expected, is that by convention, the soma marks the start of a neuron, while the soma is not the region which gets elicited; the SOCB region is. With a size of $40\mu m$ to $80\mu m$ and a clearance of $40\mu m$ to the soma, the SOCB low threshold region is significantly bigger than the soma itself (Chapter 4.2.3) and in average $100\mu m$ remote. This influence therefore, has to be considered for stimulating specific regions. The distant axon was found to have a lower stimulation threshold than the soma. This distribution of elicitability makes it difficult to estimate the threshold for a soma on a specific location [Fried et al., 2009, Nowak and Bullier, 1998].

A third obstacle in generating meaningful vision is the path of the neurons. In this simulation, the neurons were represented in a simplified form where they were assumed to have a straight and homogeneous structure, and multiple neurons are strictly parallel to each other. In human retinas the size and shape (curvature) were found to depend on the distance to the fovea, and in complex morphology. This leads to a significant difference between the stimulated vs. the perceived pattern, if multiple electrodes are used and stimulation occurs close to the fovea. Figure 4.21 depicts an example correlation between the stimulated pattern and the perceived pattern. In long-term applications of visual prosthesis, a learning effect to the new pattern has been reported [Rizzo et al., 2003].

Methods

In order to eliminate the first two obstacles, stimulation maps for different electrode spacings have been generated.

To derive the elicitability of each neuron, the soma was moved underneath each square. An incrementally rising stimulating potential was set on

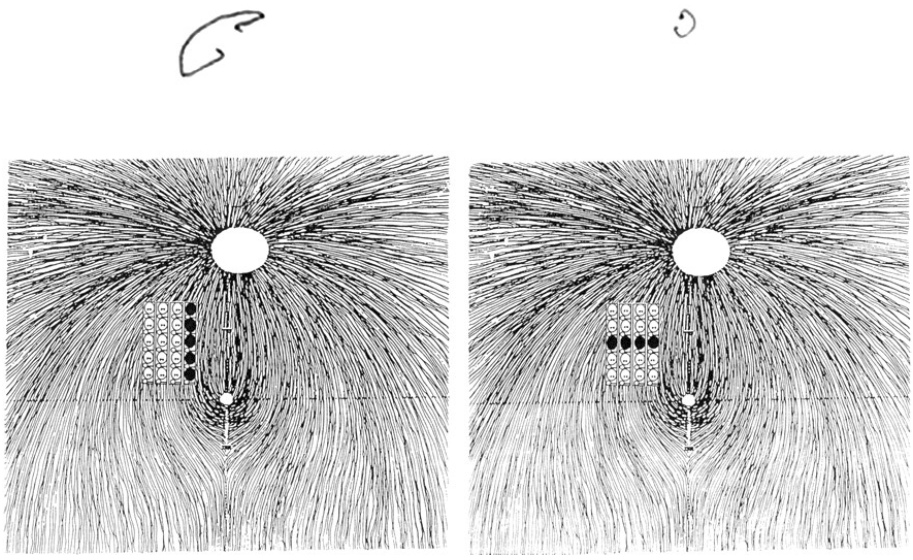


Figure 4.21: Down: 2D map of a human retina with an adjacent sub-retinal electrode array. The small white circle is the fovea centralise, the larger is the optic disc; Up: Perception caused by different stimulation pattern[Rizzo et al., 2003]

the electrodes, starting from $0V$ to $-5V$, in $100mV$ steps. The first iteration which raises a membrane potential to $40mV$ above the resting potential, is registered as activation. All these activations are visualized for each neuron, which is represented by one pixel (Figure 4.22, 4.23 and 4.23). This means that one picture represents 256, 896 and 3828, independent neurons, respectively.

The soma lies $55\mu m$ beneath the pixel under the electrode and the RGC is oriented upwards (in decreasing y direction). In Figure 4.22, one neuron is printed in light-blue (first column) to show the location applied in Figure 4.22 - Figure 4.24. Each pixel represents a tissue (spatial resolution) of 10x10μm. Therefore, the SOCB of each neuron is 4 pixels long, which is the same size as the electrode diameter.

All calculations were done for independent neurons, which means that an activated neuron does not influence its adjoining neurons. The black rings show the position of the electrodes, while the center electrode stimulates, and the outer electrodes are grounded ("guards"). The electrodes have a diameter of $40\mu m$ and are printed in scale and on the corresponding position.

Dark-blue areas are areas without a solution, either by timeout or reaching the maximum voltage. The highest simulated potential was $-5V$ and

the timeout was set to 20 seconds per pixel[3]. For computational reasons, only the left half of the image was calculated and mirrored to the right.

Stimulation potential which is significantly above the lower stimulation threshold, will stimulate a high amount of neurons in a large radius around the center electrode, and in addition, all passing electrodes where the axon is in the vicinity. All adjacent neurons which get stimulated by a specific potential have the same color.

Activation Map

The activation maps (Figure 4.22 - Figure 4.24) can provide 3 kinds of information.

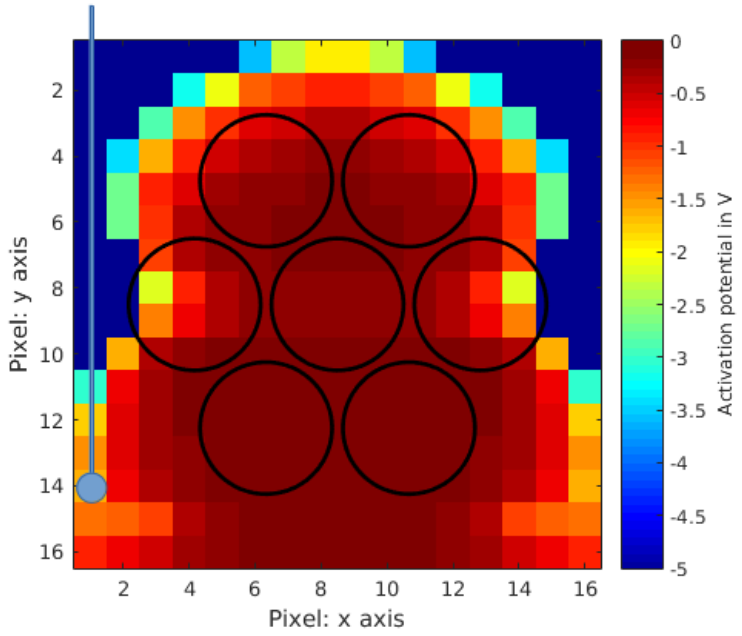


Figure 4.22: Stimulation map for 3.3μm electrode spacing, 2*128 neurons, potential resolution: 100mV For the neuron on position 1-14, the soma and axon have been visualized as well. (axon not in scale)

[3]Calculation time per image: Figure 4.22: 2.5 hours; Figure 4.23: 9.5 hours; Figure 4.24: 32 hours

Firstly, if a specific cluster of neurons is desired for stimulation, possible places for placing the electrode configuration in order to stimulate the least amount of adjoining neurons can be found. Therefore, the optimized center for the stimulated cluster is between the lower electrodes (x-axis) and amid the electrode borders of center and lower electrodes (y-axis). However, the previously mentioned effect that the most elicitable region is not directly underneath the stimulation electrode, gets stronger with increasing electrode spacing.

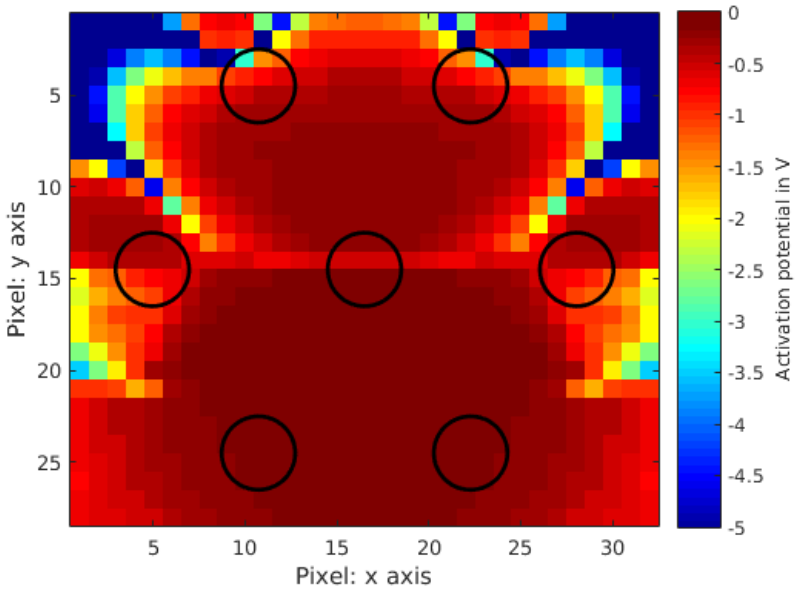


Figure 4.23: Stimulation map for 75.5μm electrode spacing, 2*448 neurons, potential resolution: 100mV

Secondly, the map provides information about the number, form and size of the stimulated neuron cluster. The largest cluster of neurons which gets stimulated with low potential, is the goblet-shaped cluster (902 neurons for 248.7μm electrode spacing and $-200mV$) under the central electrode. Additionally, there are 3 smaller clusters (around 15-20 neurons each) above the central and outer left/right electrodes. These areas can be shown best in configurations with large electrode spacings. In smaller configurations, these areas unite to form a single field. Furthermore, the SOCB shifts the region of highest elicitability beneath the lowest electrodes, but leaves it independent of the configuration, starting from the center electrode.

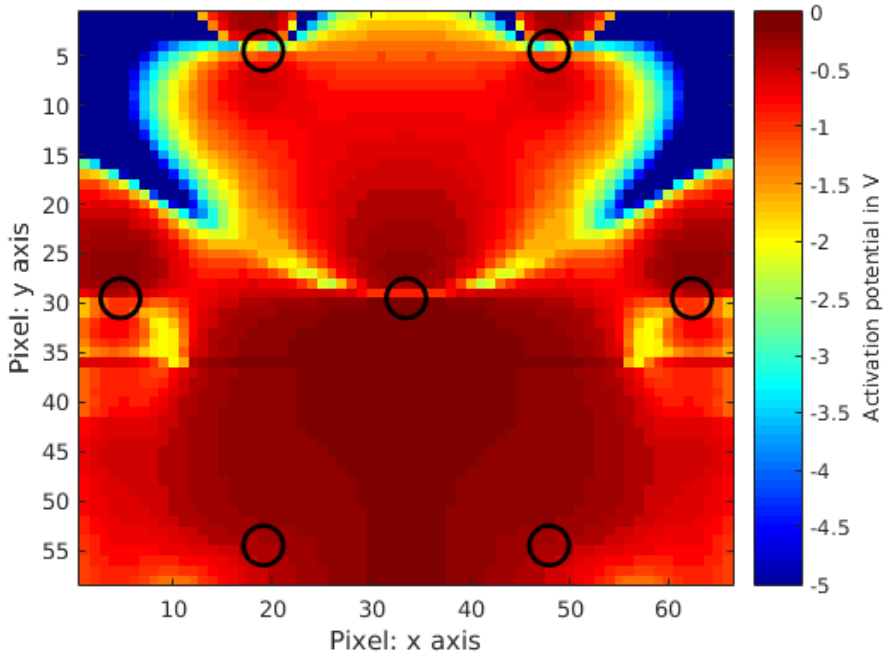


Figure 4.24: Stimulation map for 248.7μm electrode spacing, 2*1914 neurons, potential resolution: 100mV

Thirdly, these maps also show which regions are likely to become stimulated on a risen potential. In the 248.7μm electrode spacing, 904 neurons get stimulated with a small potential of $-200mV$. An increase to $-400mV$ can reach an additional 518 neurons. However, since the retina has 10 kinds of RGC, the minimal threshold might not be sufficient in all cases, otherwise additional neurons from the ones shown, might become stimulated.

All potential maps have in common, the fact that the upper electrodes might only have negligible effects on the stimulation.

Chapter 5

Discussion

In this part of the thesis, the previous results are reviewed, seen in the their full background framework.

5.1 Membrane Potential

Chapter 3 is focused on the different energy efficiencies of mammalian and amphibian RGC. Using the models of Fohlmeister et al. 1997, Fohlmeister et al. 2010 and Hodgkin & Huxley helped to capture results of intracellularly stimulated neurons. These simulations were performed using Comsol Multiphysics and MATLAB. In this chapter the two main experiments were: the energy efficiency experiment and the double-pulse experiment. In the energy efficiency experiment, the above-mentioned three models were used to calculate and compare the results. Each model simulated and analysed the absolute charge and charge density generated by the excited membrane. The main conclusion, as described in detail in Chapter 3, is: the more sophisticated mammalian soma has a partially increased energy demand than the amphibian soma. However, the overall efficiency of the mammalian cell is higher than the amphibian cell. This energy efficiency advantage might allow a faster reaction time of the more developed mammalian RGC. The second main observation of the Chapter 3 is, the double-pulse experiment with a variation of inter-pulse time. More precisely, two impulses were injected into a neuron, with varying inter-pulse time. The stimulation in both models was applied in an intra-cellular manner, within a single compartment, and with a current density of triple the minimal threshold. The purpose was to compare the inter-peak times of both Fohlmeister models in a soma. It was observed how long a neuron needs to generate an AP, after

the second injected pulse. As seen in more detail in Chapter 3, the result shows that in mammals, at small inter-pulse times and when the activating energy is not too high, there are very long inter-peak periods between the APs.

5.2 Electrode Configuration Study

In Chapter 4 the optimized center-to-center distance between stimulated electrodes was obtained. This optimization was achieved through a computer simulation in Comsol and MATLAB. More precisely, a static volume model of the retina was created in Comsol using a finite element method and afterwards this model was added into the author's framework in MATLAB. The neuronal calculation method was a compartment model where the membrane potential was derived by Fohlmeister et al. 2010. The compartments were interconnected and activated by Rattay's activating function. So, an applied potential was able to be tested to determine if sufficient to raise and propagate an action potential through the neuron. The scope of this set of calculations was to identify the maximum amount of electrodes in a retina implant, without violating the safe charge limits. The essential advantage of this set of calculations is that around a center-to-center spacing of $46.6\mu m$, an efficiency optimum appears. As shown in the above-named chapter, it is worth mentioning that a number of smaller spacings require much more energy for a proper stimulation of the neuron. A value higher than $46.6\mu m$ enlarges the system and reduces the number of potential electrodes, without adding any benefit.

Further on in Chapter 4.3, is shown which neurons are to be stimulated beneath each set of electrodes, either at minimal or higher threshold. As the main observation from this part of the thesis is that, given the position of the sodium band, the most intense stimulation happens in the triangle between the center electrode and the guarding electrodes, furthest from the optic disc.

Bibliography

Ahuja, A., Dorn, J., Caspi, A., McMahon, M., Dagnelie, G., Stanga, P., Humayun, M., Greenberg, R., et al. (2011). Blind subjects implanted with the argus ii retinal prosthesis are able to improve performance in a spatial-motor task. *British Journal of Ophthalmology*, 95(4):539–543.

Azevedo, F. A., Carvalho, L. R., Grinberg, L. T., Farfel, J. M., Ferretti, R. E., Leite, R. E., Lent, R., Herculano-Houzel, S., et al. (2009). Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain. *Journal of Comparative Neurology*, 513(5):532–541.

Baden, T., Berens, P., Bethge, M., and Euler, T. (2013). Spikes in mammalian bipolar cells support temporal layering of the inner retina. *Current Biology*, 23(1):48–52.

Behrend, M. R., Ahuja, A. K., Humayun, M. S., Chow, R. H., and Weiland, J. D. (2011). Resolution of the epiretinal prosthesis is not limited by electrode size. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 19(4):436–442.

Betts, J. (2013). *Anatomy and Physiology*. OpenStax College, Rice University, Houston, Texas.

Brash, J. (1951). *Cunningham's Textbook of Anatomy (ed. 9)*. Oxford University Press, London.

Chow, A. Y., Chow, V. Y., Packo, K. H., Pollack, J. S., Peyman, G. A., and Schuchard, R. (2004). The artificial silicon retina microchip for the treatment of visionloss from retinitis pigmentosa. *Archives of ophthalmology*, 122(4):460–469.

Coleman, P. A. and Miller, R. F. (1989). Measurement of passive membrane parameters with whole-cell recording from neurons in the intact amphibian retina. *Journal of neurophysiology*, 61(1):218–230.

Dokos, S., Suaning, G. J., and Lovell, N. H. (2005). A bidomain model of epiretinal stimulation. *IEEE transactions on neural systems and rehabilitation engineering: a publication of the IEEE Engineering in Medicine and Biology Society*, 13(2):137–146.

Finn, W. E. and LoPresti, P. G. (2010). *Handbook of Neuroprosthetic methods*. CRC Press.

Fohlmeister, J. F., Cohen, E. D., and Newman, E. A. (2010). Mechanisms and distribution of ion channels in retinal ganglion cells: Using temperature as an independent variable. *Journal of Neurophysiology*, 103(3):1357–1374.

Fohlmeister, J. F., Coleman, P., and Miller, R. F. (1990). Modeling the repetitive firing of retinal ganglion cells. *Brain research*, 510(2):343–345.

Fohlmeister, J. F. and Miller, R. F. (1997a). Impulse encoding mechanisms of ganglion cells in the tiger salamander retina. *Journal of Neurophysiology*, 78(4):1935–1947.

Fohlmeister, J. F. and Miller, R. F. (1997b). Mechanisms by which cell geometry controls repetitive impulse firing in retinal ganglion cells. *Journal of Neurophysiology*, 78(4):1948–1964.

Fried, S. I., Lasker, A. C., Desai, N. J., Eddington, D. K., and Rizzo, J. F. (2009). Axonal sodium-channel bands shape the response to electric stimulation in retinal ganglion cells. *Journal of neurophysiology*, 101(4):1972–1987.

Galbraith, B. (January 26, 2011). Neural modeling with python. Available at `http://www.neurdon.com/2011/01/26/neural-modeling-with-python-part-2/`, accessed 2016-09-30.

Gentet, L. J., Stuart, G. J., and Clements, J. D. (2000). Direct measurement of specific membrane capacitance in neurons. *Biophysical Journal*, 79(1):314–320.

Gerstner, W. and Kistler, W. M. (2002). *Spiking neuron models: Single neurons, populations, plasticity*. Cambridge university press.

Goldman, D. E. (1943). Potential, impedance, and rectification in membranes. *The Journal of general physiology*, 27(1):37–60.

Greenberg, R. J., Velte, T. J., Humayun, M. S., Scarlatis, G. N., and de Juan Jr, E. (1999). A computational model of electrical stimulation of the retinal ganglion cell. *Biomedical Engineering, IEEE Transactions on*, 46(5):505–514.

Guo, T., Tsai, D., Bai, S., Morley, J. W., Suaning, G. J., Lovell, N. H., and Dokos, S. (2014). Understanding the retina: A review of computational models of the retina from the single cell to the network level. *Critical Reviews in Biomedical Engineering*, 42(5):419–436.

Hodgkin, A. L. and Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4):500–544.

Humayun, M. S., Weiland, J. D., Fujii, G. Y., Greenberg, R., Williamson, R., Little, J., Mech, B., Cimmarusti, V., Van Boemel, G., Dagnelie, G., et al. (2003). Visual perception in a blind subject with a chronic microelectronic retinal prosthesis. *Vision research*, 43(24):2573–2581.

Jeng, J., Tang, S., Molnar, A., Desai, N., and Fried, S. (2011). The sodium channel band shapes the response to electric stimulation in retinal ganglion cells. *Journal of neural engineering*, 8(3):036022.

Jensen, R. J. and Rizzo, J. F. (2008). Activation of retinal ganglion cells in wild-type and rd1 mice through electrical stimulation of the retinal neural network. *Vision research*, 48(14):1562–1568.

Jensen, R. J., Rizzo, J. F., Ziv, O. R., Grumet, A., and Wyatt, J. (2003). Thresholds for activation of rabbit retinal ganglion cells with an ultrafine, extracellular microelectrode. *Investigative ophthalmology & visual science*, 44(8):3533–3543.

Joarder, S. A., Abramian, M., Suaning, G. J., Lovell, N. H., and Dokos, S. (2011). A continuum model of retinal electrical stimulation. *Journal of Neural Engineering*, 8(6):066006.

Kahle, W. (1991). *dtv-Atlas der Anatomie : Tafeln und texte*. Taschenatlas der Anatomie; für Studium und Praxis. Deutscher Taschenbuch Verlag.

Karwoski, C. J., Frambach, D., and Proenza, L. M. (1985). Laminar profile of resistivity in frog retina. *Journal of neurophysiology*, 54(6):1607–1619.

Karwoski, C. J. and Xu, X. (1999). Current source-density analysis of light-evoked field potentials in rabbit retina. *Visual neuroscience*, 16(2):369–377.

Keat, J., Reinagel, P., Reid, R. C., and Meister, M. (2001). Predicting every spike: a model for the responses of visual neurons. *Neuron*, 30(3):803–817.

Kolb, H., Fernandez, E., and Nelson, R. (1995). Visual acuity–webvision: The organization of the retina and visual system. Available at `http://webvision.med.utah.edu/`, accessed 2016-12-19.

Kurowski, P. (1994). Avoiding pitfalls in fea. *Machine Design*, 66.

Lipton, S. A. and Tauck, D. L. (1987). Voltage-dependent conductances of solitary ganglion cells dissociated from the rat retina. *The Journal of Physiology*, 385(1):361–391.

Litke, A., Bezayiff, N., Chichilnisky, E., Cunningham, W., Dabrowski, W., Grillo, A., Grivich, M., Grybos, P., Hottowy, P., Kachiguine, S., et al. (2004). What does the eye tell the brain?: Development of a system for the large-scale recording of retinal output activity. *Nuclear Science, IEEE Transactions on*, 51(4):1434–1440.

Margalit, E., Maia, M., Weiland, J. D., Greenberg, R. J., Fujii, G. Y., Torres, G., Piyathaisere, D. V., O'Hearn, T. M., Liu, W., Lazzi, G., et al. (2002). Retinal prosthesis for the blind. *Survey of ophthalmology*, 47(4):335–356.

Mathieson, K., Kachiguine, S., Adams, C., Cunningham, W., Gunning, D., Shea, V., Smith, K., Chichilnisky, E., Litke, A., Sher, A., et al. (2004). Large-area microelectrode arrays for recording of neural signals. *Nuclear Science, IEEE Transactions on*, 51(5):2027–2031.

Merrill, D. R., Bikson, M., and Jefferys, J. G. (2005). Electrical stimulation of excitable tissue: design of efficacious and safe protocols. *Journal of neuroscience methods*, 141(2):171–198.

Nowak, L. and Bullier, J. (1998). Axons, but not cell bodies, are activated by electrical stimulation in cortical gray matter i. evidence from chronaxie measurements. *Experimental brain research*, 118(4):477–488.

Oh, S. M. et al. (1996). Enzyme sensors prepared by electrodeposition on platinized platinum electrodes. *Electrochimica Acta*, 41(15):2433–2439.

Palanker, D., Vankov, A., Huie, P., and Baccus, S. (2005). Design of a high-resolution optoelectronic retinal prosthesis. *Journal of neural engineering*, 2(1):105–120.

Pfützner, H. (2012). *Angewandte Biophysik*. Springer-Verlag.

Pointer, J. (2004). Understanding accuracy and discretization error in an fea model. *ANSYS 7.1, 2004 Conference, Wooward Governor Company*.

Rao, C., Yuan, X.-H., Zhang, S.-J., Wang, Q.-L., and Huang, Y.-S. (2008). epiretinal prosthesis forouter retinal degenerative diseases. *International Journal of Ophthalmology*, 1(3):273–276.

Rattay, F. (1990). *Electrical nerve stimulation*. Springer.

Rattay, F. (1999). The basic mechanism for the electrical stimulation of the nervous system. *Neuroscience*, 89(2):335 – 346.

Rattay, F. and Danner, S. M. (2014). Peak i of the human auditory brainstem response results from the somatic regions of type i spiral ganglion cells: Evidence from computer modeling. *Hearing research*, 315:67–79.

Rattay, F., Greenberg, R., and Resatz, S. (2002). Neuron modeling. *Handbook of Neuroprosthetic Methods*.

Rattay, F. and Resatz, S. (2004). Effective electrode configuration for selective stimulation with inner eye prostheses. *Biomedical Engineering, IEEE Transactions on*, 51(9):1659–1664.

Resatz, S. (2005). *Analysis of retinal cell excitation with visual neuroprostheses*. Ph.D. thesis, Vienna University of Technology.

Rizzo, J. F., Wyatt, J., Loewenstein, J., Kelly, S., and Shire, D. (2003). Perceptual efficacy of electrical stimulation of human retina with a microelectrode array during short-term surgical trials. *Investigative ophthalmology & visual science*, 44(12):5362–5369.

Schanze, T., Wilms, M., Eger, M., Hesse, L., and Eckhorn, R. (2002). Activation zones in cat visual cortex evoked by electrical retina stimulation. *Graefe's archive for clinical and experimental ophthalmology*, 240(11):947–954.

Sekirnjak, C., Hottowy, P., Sher, A., Dabrowski, W., Litke, A. M., and Chichilnisky, E. J. (2006). Electrical stimulation of mammalian retinal ganglion cells with multielectrode arrays. *Journal of Neurophysiology*, 95(6):3311–3327.

Seung, H. S. and Sümbül, U. (2014). Neuronal cell types and connectivity: lessons from the retina. *Neuron*, 83(6):1262–1272.

Shah, C. (2002). Mesh discretization error and criteria for accuracy of finite element solutions. *Ansys Users Conference*, page 12.

Weiland, J. D., Humayun, M., Dagnelie, G., de Juan Jr, E., Greenberg, R. J., and Iliff, N. T. (1999). Understanding the origin of visual percepts elicited by electrical stimulation of the human retina. *Graefe's archive for clinical and experimental ophthalmology*, 237(12):1007–1013.

Wilke, R., Moghadam, G. K., Lovell, N., Suaning, G., and Dokos, S. (2011). Electric crosstalk impairs spatial resolution of multi-electrode arrays in retinal implants. *Journal of neural engineering*, 8(4):046016.

Wohrer, A. and Kornprobst, P. (2009). Virtual retina: a biological retina model and simulator, with contrast gain control. *Journal of computational neuroscience*, 26(2):219–249.

Yanai, D., Weiland, J. D., Mahadevappa, M., Greenberg, R. J., Fine, I., and Humayun, M. S. (2007). Visual performance using a retinal prosthesis in three subjects with retinitis pigmentosa. *American journal of ophthalmology*, 143(5):820–827.

Yue, L., Falabella, P., Christopher, P., Wuyyuru, V., Dorn, J., Schor, P., Greenberg, R. J., Weiland, J. D., and Humayun, M. S. (2015). Ten-year follow-up of a blind patient chronically implanted with epiretinal prosthesis argus i. *Ophthalmology*, 122(12):2545–2552.

Zrenner, E. (2014). Restoring vision to the blind: The new age of implanted visual prostheses. *Translational Vision Science and Technology*, 3(7):3.

Zrenner, E., Bartz-Schmidt, K. U., Benav, H., Besch, D., Bruckmann, A., Gabel, V.-P., Gekeler, F., Greppmaier, U., Harscher, A., Kibbel, S., et al. (2011). Subretinal electronic chips allow blind patients to read letters and combine them to words. *Proceedings of the Royal Society of London B: Biological Sciences*, 278(1711):1489–1497.

Appendix A

Resources

While this section can only contain a few important classes of the code, all resources and source codes, used in this thesis, are available on GitHub `https://github.com/plejaden/RGC_Stimulation` . The sourcecode of this thesis comprising 6965 written and additionally 6172 Comsol-generated lines of code in 61 files[1]. It is entirely written in MATLAB, but a connected Comsol Multiphysics software is necessary to run it, despite the fact that the finite element model is created in Comsol and exported to MATLAB. Anyway, this exported files can be found on GitHub too.

A.1 Sourcecode

[1] counted by cloc `http://cloc.sourceforge.net`

```
1   classdef mySimulationBuilder
2       % Author: Watzinger Anton
3       % Last edited: 2017-02-11
4
5       %% Start with:
6       % clear; mySimulationBuilder
    ('fcn2010_cat_20deg',1,'2electrodes_epiretinal',100);
7       methods
8           % class constructor
9           function this = mySimulationBuilder(simulationModelName,
    potentialMultiplicator, electrodeConfiguration,compartmentNumbers)
10
11              global error;
12              error = ''; % no error occured
13              starttime=datestr(now, 'yyyy-mm-dd HH:MM:SS');
14              display(' ');
15              display([ 'Simulation started at ' starttime]);
16              close all;
17              warning ('off','all');
18
19              tic;
20              numberOfParameters=nargin();
21
22              if(numberOfParameters <= 3)
23                  display('Quick-guide for using this system');
24                  display('mySimulationBuilder
    (''fcn2010_cat'',1,''2electrodes_epiretinal'',100)');
25                  display(' ');
26                  display('#1 parameter: used model for calculation.
    Implemented models are:');
27                  dir('membraneModels/*.m')
28                  display(' ');
29                  display('#2 parameter: voltage multiplier. use 1 to use the
    exact volatge from the comsol model');
30                  display(' ');
31                  display('#3 parameter: Geometry. Implemented geometries
    are:');
32                  this.displayPhysicModels();
33                  display(' ');
34                  display('#4 parameter: Number of compartments. use a value
    from 10 to 70');
35                  return;
36              end % if(numberOfParameters <= 3)
37
38
39              display('available Geometries:');
40              this.displayPhysicModels();
41
42              if(numberOfParameters == 4)
43
44                  % create retina, nerve and electrode geometry, mesh, run
    physics %
45                  if (isempty (error))
46                      display('## calculate finite element potentials ##'); toc;
47                      geometryProperties=this.organizeModel
    (this,electrodeConfiguration,compartmentNumbers,potentialMultiplicator);
48                      geometryProperties.filename=[ simulationModelName,'+',
    num2str(potentialMultiplicator),'+', electrodeConfiguration,'+',num2str
    (compartmentNumbers),'+',datestr(now, 'yyyy-mm-dd_HH:MM:SS') ];
49                  end
50
51                  % envokes the nerve model for calculating the membrane
    potential
52                  if (isempty (error))
53                      simulationModel = this.envokeSimulationModel
    (simulationModelName, geometryProperties);
54                  end %  if (isempty (error))
55
56                  %Calculate the model
```

```
57                  if (isempty (error))
58                      display(' '); display('## solveing ode ##'); toc;
59                      [T, Y] = simulationModel.run(geometryProperties);
60                  end % if (isempty (error))
61
62                  if (isempty (error))
63                      % check if there are some differences in the resulting
    matrix, which may be an AP %
64                      if (max(max(Y(:,1,:)))<min(min(Y(:,1,:)))+50)
65                          display('WARNING: No AP spike detected!'); toc;
66                      end % if (max(max(Y(:,1,:)))<min(min(Y(:,1,:)))+50)
67                  end % if (isempty (error))
68
69                  %Plot the results
70                  global currentObserver;
71                  global activation;
72                  filename=['output/',geometryProperties.filename,'.mat'];
73                  save (filename);
74
75                  sliderPlot(T,Y,geometryProperties);
76
77              elseif (numberOfParameters > 4)
78                  %disp('checkpoint');
79                  fprintf('%d parameters, which is too much (3 parameters
    needed: Model, impulse Type and electrodeConfiguration)',nargin);
80              end %if (nargin == 3)
81              display(' '); toc;
82              display(' ');
83              display(' ');
84          end %function myModelBuilder
85
86      end %methods
87
88
89      methods (Static)
90          function geometryProperties=organizeModel
    (this,modelType,compartmentNumbers,potentialMultiplicator)
91
92              geometryProperties.potMultiplicator=potentialMultiplicator;
93              geometryProperties.numberOfCompartments=compartmentNumbers;
94              geometryProperties.iStim=0;
95              geometryProperties.rho = 70.6/1000; % 70.6 Ohm/um => Ohm/m
96              geometryProperties.rhoLayer=18.2/1000; % 18.2 Ohm/um => Ohm/m
97
98              %% Values, generated without geometry.
99              % stimulate with already measured values
100             if strfind(modelType, 'prestored')
101                 display(' %% Using stored geometry results');
102                 load (['retinaElectrodeModels/' 'preRenderedModels/'
    modelType '.mat'])
103
    geometryProperties.externalVoltages=geometryProperties.externalVoltages*potentialMultiplica
104                 geometryProperties.compFeatures = [ repmat({'Soma'},1,1) ;
    repmat({'IS'},3,1); repmat({'SOCB'},3,1); repmat({'Axon'},93,1) ];
105                 geometryProperties.compartmentRadius=[ repmat(10*10^-6,1,1) ;
    repmat(10^-6,3,1); repmat(10^-6,3,1); repmat(10^-6,93,1) ];
106                 return;
107             end
108
109             %stimulate with dirac
110             if strcmpi(modelType,'testPointStimulation')
111                 display('TEST-MODE: Using Point function');
112                 geometryProperties.model='noModel-precalculated Values only';
113                 geometryProperties.numberOfCompartments=compartmentNumbers;
114                 geometryProperties.electrodeDistance=1e-04;
115                 geometryProperties.compartmentRadius=5e-03;
116                 geometryProperties.compartmentRadius=1e-05; % = 10um %
    usually 0.2-20um
117                 geometryProperties.nerveLength=3e-3;
```

```
118
    geometryProperties.distanceBetrweenCompartments=geometryProperties.nerveLength/
    compartmentNumbers; %m
119
120                 geometryProperties.compartmentRadius=5e-05; %m
121
122                 oneComp=zeros(geometryProperties.numberOfCompartments,1);
123                 oneComp(6)=1;
124                 geometryProperties.externalVoltages=oneComp;
125
    geometryProperties.externalVoltages=geometryProperties.externalVoltages*1000;
    %convert volts to mV
126
127                 return;
128             end
129
130             % no external stimulation
131             if strcmpi(modelType,'noStimulation')
132                 display('TEST-MODE: Disabled all stimulus');
133                 geometryProperties.model='noModel-precalculated Values only';
134                 geometryProperties.rho = 0.0706;
135                 geometryProperties.numberOfCompartments=compartmentNumbers;
136
137                 geometryProperties.compartmentRadius=[ repmat(10^-6,100,1) ];
138                 geometryProperties.electrodeDistance=6*10^-5;
139                 geometryProperties.nerveLength=3e-3;
140
    geometryProperties.distanceBetrweenCompartments=geometryProperties.nerveLength/
    compartmentNumbers; %m
141
142                 geometryProperties.externalVoltages=zeros
    (1,geometryProperties.numberOfCompartments);
143                 return;
144
145             end % if strcmpi(modelType,'noStimulation')
146
147             % no external stimulation
148             if strcmpi(modelType,'intercellularStimulationOnly')
149                 display('_-_-_-_-_-_-_')
150                 display('INTERCELLULAR-STIMULATION:');
151                 display('Disabled all stimulus but the
    "potentialMultiplicator" [ua/cm^2] at the first compartment');
152                 geometryProperties.model='noModel-virtual soma stimulaton
    only';
153                 geometryProperties.rho = 50.5;
154                 geometryProperties.numberOfCompartments=compartmentNumbers;
155                 geometryProperties.iStim=potentialMultiplicator;
156
157                 geometryProperties.compFeatures = [ repmat({'Soma'},1,1) ;
    repmat({'IS'},3,1); repmat({'SOCB'},3,1); repmat({'Axon'},93,1) ];
158                 geometryProperties.compartmentRadius=[ repmat(10*10^-6,1,1) ;
    repmat(10^-6,3,1); repmat(10^-6,3,1); repmat(10^-6,93,1) ];
159                 geometryProperties.electrodeDistance=6*10^-5;
160                 geometryProperties.nerveLength=3e-3;
161
    geometryProperties.distanceBetrweenCompartments=geometryProperties.nerveLength/
    compartmentNumbers; %m
162
163                 geometryProperties.externalVoltages=zeros
    (1,compartmentNumbers);
164                 return;
165
166             end % if strcmpi(modelType,'intercellularStimulationOnly')
167
168
169             % no external stimulation
170             if strcmpi(modelType,'intercellularSinglecompStimulationOnly')
171                 display('_-_-_-_-_-_-_')
172                 display('INTERCELLULAR-STIMULATION (SINGLE COMP):');
```

```
173                 display('Disabled all stimulus but the
    "potentialMultiplicator" [ua/cm^2] at the first compartment');
174                 geometryProperties.model='noModel-virtual soma stimulaton
    only';
175                 geometryProperties.rho = 50.5;
176                 geometryProperties.numberOfCompartments=1;
177                 geometryProperties.iStim=potentialMultiplicator;
178
179                 geometryProperties.compFeatures =  {'Soma'} ;
180                 geometryProperties.compartmentRadius= (5*10^-6) ;
181                 geometryProperties.electrodeDistance=6*10^-5; % not used
182                 geometryProperties.distanceBetrweenCompartments=30*10^-6; %m
183
184                 geometryProperties.externalVoltages=zeros(1,1);
185                 return;
186             end %   if strcmpi
    (modelType,'intercellularSinglecompStimulationOnly')
187
188             %% Usage of real geometry
189             geoFuncName=['geometry_' modelType ];
190             gdescName=['description_' modelType ];
191
192             if (exist(geoFuncName, 'file') && exist(gdescName, 'file') )
193                 gmodel=str2func(geoFuncName);
194                 gdesc=str2func(gdescName);
195             else
196                 display('Geometry: no correct geometry specified, using a
    very simple geometry');
197                 gmodel=simple2ElectrodeConfiguration();
198                 gdesc=description_simple2ElectrodeConfiguration();
199             end % if (exist(geoFuncName, 'file') && exist(gdescName, 'file') )
200
201             % envoking the gmodel and gdesc functionpointers.
202             geometryProperties.model=gmodel();
203             geometryProperties.desc=gdesc(); toc;
204
205             geometryProperties.numberOfCompartments=compartmentNumbers;
206             geometryProperties=this.getExVoltages(geometryProperties); % add
    most of the properties
207         end % geometryProperties=organizeModel
    (this,modelType,compartmentNumbers,potentialMultiplicator)
208
209         %% get the external Voltages form the geometry
210         function geometryProperties=getExVoltages(geometryProperties)
211
212             retinaModel=geometryProperties.model;
213             compartmentNumbers=geometryProperties.numberOfCompartments;
214             geometryProperties.distanceBetrweenCompartments=4; % ??
215
216             import com.comsol.model.*
217             import com.comsol.model.util.*
218
219             %% Step 1: get the names of all retinal geometries
220             nerveGeometryNames=cellstr
    ( geometryProperties.desc.neuronPartNames(:,1) );
221
    electrodeGeometryNames=geometryProperties.desc.electrodeGeometryNames;
222
223             % get variable definition, if there are any
224             if max(strcmp(fieldnames(geometryProperties.desc),'variables'))
225                 for i=1:length(geometryProperties.desc.variables)
226                     evalc(char(geometryProperties.desc.variables(i)));
227                 end
228             end
229
230             %% Step 2: get the coordinates from the geometries
231             numberOfNeuronparts= size(nerveGeometryNames(:,1),1);
232             neuronCoords.x=zeros(numberOfNeuronparts,1);
233             neuronCoords.y=zeros(numberOfNeuronparts,1);
```

```
234             neuronCoords.z=zeros(numberOfNeuronparts,1);
235             neuronCoords.h=zeros(numberOfNeuronparts,1);
236             neuronCoords.r=zeros(numberOfNeuronparts,1);
237             neuronCoords.name=cell(numberOfNeuronparts,1);
238
239             for i=1:numberOfNeuronparts
240                 neuronPartName=nerveGeometryNames(i,:);
241                 propertiesNerve=mphgetproperties(retinaModel.geom
    ('geom1').feature(neuronPartName));
242                 display(['reading neuronal part: ',neuronPartName]);
243                 neuronCoords.name{i}=neuronPartName;
244                 neuronCoords.x(i)=eval(propertiesNerve.x);
245                 neuronCoords.y(i)=eval(propertiesNerve.y);
246                 neuronCoords.z(i)=eval(propertiesNerve.z);
247
248                 cylPresent = find(cellfun('length',regexp
    (neuronPartName,'cyl')) == 1);
249                 if (cylPresent)
250                     neuronCoords.r(i)=eval(propertiesNerve.r);
251                     neuronCoords.h(i)=eval(propertiesNerve.h);
252                 end %if (cylPresent)
253
254                 sphPresent = find(cellfun('length',regexp
    (neuronPartName,'sph')) == 1);
255                 if (sphPresent)
256                     neuronCoords.r(i)=eval(propertiesNerve.r);
257                     neuronCoords.h(i)=eval(propertiesNerve.r);
258                 end %if (sphPresent)
259
260             end % for i=1:numberOfNeuronparts
261
262             xNeuronStart = max(neuronCoords.x);  yNeuronStart = max
    ( neuronCoords.y); zNeuronStart = max(neuronCoords.z);
263             radiusNerve = neuronCoords.r; heightNerve=sum( neuronCoords.h);
264             geometryProperties.nerveLength=heightNerve;
265
266             % extract the electrode geometry
267             numberOfElectrods=size(electrodeGeometryNames);
268             numberOfElectrods=numberOfElectrods(1,1);
269             electrodeCoords.x=zeros(numberOfElectrods,1);
270             electrodeCoords.y=zeros(numberOfElectrods,1);
271             electrodeCoords.z=zeros(numberOfElectrods,1);
272             electrodeCoords.radius=zeros(numberOfElectrods,1);
273             electrodeCoords.height=zeros(numberOfElectrods,1);
274
275             for i=1:numberOfElectrods
276                 electrodeName=electrodeGeometryNames(i,:);
277                 propertiesElectrode=mphgetproperties(retinaModel.geom
    ('geom1').feature(electrodeName));
278                 electrodeCoords.x(i)=eval(propertiesElectrode.x);
279                 electrodeCoords.y(i)=eval(propertiesElectrode.y);
280                 electrodeCoords.z(i)=eval(propertiesElectrode.z);
281                 if (cell2mat(strfind(electrodeName,'cyl')))
282                     electrodeCoords.radius(i)=eval(propertiesElectrode.r);
283                     electrodeCoords.height(i)=eval(propertiesElectrode.h);
284                 end %(cell2mat(strfind(electrodeName,'cyl')))
285             end % i=1:numberOfElectrods
286
287             %% Step 4:
288             % find out how the geometry is arranged (how many parallel
    nerves) %
289             nrOfNerveStrains=1; % default: 1
290
291             %% Step 5:
292             % Calculate the pointes of measurement
293             geometryProperties.electrodeDistance=abs(zNeuronStart-min
    (electrodeCoords.z));
294             compartmentPerStrain=compartmentNumbers/nrOfNerveStrains;
295             delta=heightNerve/(compartmentNumbers-1);
```

```
296             measurementPoints=zeros(compartmentPerStrain*nrOfNerveStrains,4);
297
298             thisZ=zNeuronStart;
299             for i=1:length(measurementPoints)-1
300                 thisX = xNeuronStart;
301                 thisY = min(neuronCoords.y)+(i-1)*delta;
302                 thisZ = findZinGeometry(thisX,thisY, neuronCoords,thisZ);
303
304                 measurementPoints(i,:)=[1,thisX,thisY,thisZ];
305             end %for i=1:length(measurementPoints)-1
306             measurementPoints(compartmentNumbers,:)=[1,xNeuronStart,min
    (neuronCoords.y)+(compartmentNumbers)*delta,zNeuronStart];
307
308             % multi-neuron in row 1, but currently not supported.
309             measurementPoints(:,1)=[];
310
311             %% Step 6
312             %get the corresponding values from the FEM model
313
314             %Syntax: 3xn => 1xn
315             % x1 x2 ...
316             % y1 y2 ... => V1 v2
317             % z1 z2 ...
318             voltageArray=mphinterp
    (retinaModel,'V','coord',measurementPoints');
319             if  (isempty (voltageArray))
320                 error='can not load external potential. Abort.';
321                 disp(error);
322                 return;
323             end % (isempty (voltageArray))
324
325             geometryProperties.distanceBetrweenCompartments=delta;
326             voltageArray=voltageArray*geometryProperties.potMultiplicator;
327             geometryProperties.externalVoltages=voltageArray *1000; %V => mV
328
329             %% Step 7: identify feature for each compartment
330             geometryProperties.compFeatures=createCompNeuronTypeVector
    (geometryProperties,measurementPoints,neuronCoords);
331             geometryProperties.compartmentRadius=createNeuronRadiusVector
    (geometryProperties,radiusNerve);
332
333         end
334
335         %% Chose and start a membrane model (e.g. hodgkin huxley)
336         function thismodel=envokeSimulationModel(simulationModel,
    geometryProperties)
337             if (~exist(simulationModel, 'file'))
338                 error='Model not implemented';
339                 disp(error);
340                 return;
341             end % if (~exist(simulationModel, 'file'))
342
343             % Create a model-Handler of parametered simulation-model
344             modelFunctionhandler = str2func(simulationModel);
345             thismodel=modelFunctionhandler(geometryProperties);
346
347             % Check if model is an isa (InStAnce of) the abstract model
348             if (~isa(thismodel,'basicModel'))
349                 error='No model loaded. exiting';
350                 disp(error);
351                 return;
352             else
353                 display(['Using ',simulationModel, ' model']);
354             end %if (~isa(thismodel,'abstractModel'))
355
356         end % function envokeSimulationModel
357
358         %% display all available physical imput geometries on console
359         function displayPhysicModels()
```

```
360
361             listOfImplementedModelsStruct=dir('src/retinaElectrodeModels/
    geometry*.m');
362             for i=1:length(listOfImplementedModelsStruct)
363                 listOfImplementedModelsStruct(i).geoName=strrep
    (listOfImplementedModelsStruct(i).name,'geometry_','');
364                 listOfImplementedModelsStruct(i).geoName=strrep
    (listOfImplementedModelsStruct(i).geoName,'.m','');
365                 display(listOfImplementedModelsStruct(i).geoName);
366             end
367             display(' ');
368             display('or in addition, the following test parameters are
    available: ');
369             display('testValues');
370             display('testPointStimulation');
371             display('NoStimulation');
372             display('somaStimulationOnly');
373             display(' ');
374         end %function displayPhysicModels()
375
376     end %methods (Static)
377 end %classdef
```

```
classdef basicModel
    % Author: Watzinger Anton
    % Last edited: 2017-02-11

    %% Abstract model for definition of MembraneModels
    properties (Abstract)
        iPause;
        iStim; %(uA/cm^2) amount of current stimulation for iStimTime
        iStimTime; %(ms) time of current stimulation
        simulationTime; %(s)
        startParameters;
        numberOfStateVariables;
        Ve; r; area; volume;
        ohmAx;

        %performance analysis
        t0;
        timeout;
    end % properties (Abstract)

    properties (Abstract = false)
        % define the properties of the class here, (like fields of a struct)

    end %(Abstract = false)

    methods (Abstract=true)

        [result] = h0(this);
        [result] = m0(this);
        [result] = n0(this);

        [result] = differentialEq(this,t,y);

        [T, Y] = postProcessing(this,T,Y);

    end %methods (Abstract=true)

    %%
    methods (Abstract = false)
        function [T, Y] = run(this,geometryProperties)
            clear 'currentObserver'; global currentObserver;

            % timing analysis
            this.t0=clock;

            %% current analysis
            currentObserver.compartment=1; currentObserver.time = 0;
currentObserver.INa = 0; currentObserver.IK = 0;
currentObserver.externalVoltage=geometryProperties.externalVoltages;
            currentObserver.IKA = 0; currentObserver.IKCa = 0;
currentObserver.ICa = 0; currentObserver.IL = 0;
currentObserver.stimulusCurrent = 0; currentObserver.externalCurrentSet=false;

            % rotate the external voltages, if they are not in array shape
            voltageForm=size(geometryProperties.externalVoltages);
            if (voltageForm(1)<voltageForm(2))

geometryProperties.externalVoltages=geometryProperties.externalVoltages';
            end
            global activation;
            activation.current=zeros(max(voltageForm),1);
            activation.activatingFunction=0;
            activation.time=0;
            activation.activatingFunctionCompartment=56;

            % rho = R * q / l => R = L * rho / q
            rhoFiber = this.rhoFiber*10^-3; %[Ohm*mm]
            compLength=geometryProperties.distanceBetrweenCompartments; % [m]
            compRad=geometryProperties.compartmentRadius; % [m]
```

```
65              compNumb=geometryProperties.numberOfCompartments;
66              thisSurface=2.*compRad.*pi().*compLength+2.*compRad.*compRad.*pi
    ();
67              display(['compLength:', num2str(compLength)]);
68              display(['compSurface (first cmp): ', num2str(thisSurface(1)), '
    m^2']);
69              display(['compSurface (first cmp): ', num2str(thisSurface(1)*
    (1000*1000)), ' mm^2']);
70              display(['compSurface (first cmp): ', num2str(thisSurface(1)*
    (1000*1000*1000*1000)), ' um^2']);
71
72              l_a=(compLength * 2); % [m]
73              A_a=(compRad .* 1000).^2*pi; % [mm^2]
74
75              this.ohmAx= rhoFiber*l_a./A_a; % ( [Ohm*mm] ) * m / (mm^2) =
    [kOhm]
76
77              % save the geometry Properties to external file: very useful for
    debugging %
78              geometryProperties.modelproperties=this;
79
80              simTime=[0 this.simulationTime];
81
82              try
83                  %% solve the equation
84                  % preferred solver: ode113
85                  %             [T, Y] = ode15s(@(t,y) differentialEq(this, t, y
    ), simTime, this.startParameters); % stiff
86                  %             [T, Y] = ode23(@(t,y) differentialEq(this, t, y
    ), simTime, this.startParameters); % non stiff
87                  %             [T, Y] = ode23s(@(t,y) differentialEq(this, t, y
    ), simTime, this.startParameters); % stiff
88                  %             [T, Y] = ode45(@(t,y) differentialEq(this, t, y
    ), simTime, this.startParameters);% non stiff
89                  [T, Y] = ode113(@(t,y) differentialEq(this, t, y ), simTime,
    this.startParameters);% stiff
90              catch
91                  display('EXCEPTION: ');
92                  display('Error: No solution detected');
93                  global exceptionValue;
94
95                  Y=exceptionValue;
96                  T=zeros(compNumb);
97
98              end
99
100             %%
101             if (~isreal(Y))
102                 disp('Y is complex, this should not happen!!!');
103                 return;
104             end
105
106             if (size(Y,2) <=compNumb)
107                 disp('Y does not contain gating information, this should not
    happen!!!');
108                 return;
109             end
110
111             % Y is an Array (Timesteps x (compartments * variables) e.g. 4612
    x 88 %
112             % But we want: Timesteps x variables x compartments e.g. 4612 x 8
    x 11 %
113             % slice up the array
114             calcPointsPerTime=max(size(Y(1,:))); % hols: |comparment| * |
    statvariables|%
115             compL=calcPointsPerTime/(this.numberOfStateVariables+1);
116             timeSlices=length(T);
117             tempY=zeros(timeSlices,this.numberOfStateVariables+1, compL);
118
```

```
119             for i=1:this.numberOfStateVariables+1;
120                 tempY(:,i,:)=Y(:,(i-1)*compL+1:i*compL);
121             end % for i=1:this.numberOfStateVariables+1;
122             Y=tempY;
123 
124             % do some post-processing
125             [T, Y]=this.postProcessing(this,T,Y);
126 
127         end %function [T, Y] = run(this)
128     end %methods (Abstract = false)
129 
130     methods (Static)
131 
132         %% numberOfCompartments [1]; Ve [mV]; pulseMagnification [1]
133         function stimulusCurrentExtracomp = compartmentStim
    (this,numberOfCompartments, Ve, pulseMagnification)
134             global currentObserver
135             stimulusCurrentExtracomp=zeros(numberOfCompartments,1);
136             if (pulseMagnification~=0)
137 
138                 %first compartment
139                 if (this.iStim==0 )
140                     deltaVEfirst=Ve(1)-Ve(2); % mV
141                     firstOhmAx=(this.ohmAx(1)+this.ohmAx(2))./2;
142                     stimulusCurrentExtracomp(1) = deltaVEfirst./
    (firstOhmAx.*2);  % mV / kOhm = uA
143                 end % (iStim~=0)
144 
145                 % middle compartments
146                 deltaVE1=Ve(1:numberOfCompartments-2)-Ve
    (2:numberOfCompartments-1);
147                 deltaVE2=Ve(3:numberOfCompartments)-Ve
    (2:numberOfCompartments-1); % the delta values, shifted at once.
148                 centerOhmAx=(this.ohmAx(1:numberOfCompartments-2)+this.ohmAx
    (2:numberOfCompartments-1))/2';
149                 stimulusCurrentExtracomp(2:numberOfCompartments-1) =
    (deltaVE1 + deltaVE2)./(centerOhmAx');  % mV / kOhm = uA ; the vector is
    rotated automatically %
150 
151                 % last compartment
152                 deltaVElast=Ve(numberOfCompartments-1)-Ve
    (numberOfCompartments);
153                 lastOhmAx=this.ohmAx(numberOfCompartments-1)+this.ohmAx
    (numberOfCompartments);
154                 stimulusCurrentExtracomp(numberOfCompartments) = deltaVElast./
    (lastOhmAx);  % mV / kOhm = uA
155                 stimulusCurrentExtracomp =
    stimulusCurrentExtracomp.*pulseMagnification;
156 
157                 if (currentObserver.externalCurrentSet==false)
158                     currentObserver.externalVoltage=Ve;
159                     currentObserver.externalCurrent=stimulusCurrentExtracomp;
160                     currentObserver.externalCurrentSet=true;
161                 end
162 
163             end % if (pulseMagnification~=0)
164 
165         end % function compartmentStim
166 
167 
168         function X = rStim(pause,current)
169             %function X = rStim(time, pulsl,pauseBeforeStim, current)
170             % X = (time>pauseBeforeStim) & (time<(pauseBeforeStim+pulsl)); %
    0 or 1
171             % X=X*current;
172             if (pause~=0)
173                 X=current;
174             else
175                 X=0;
```

```
176             end
177
178         end % function rStim
179
180
181         %% current calculations
182         function stimulusCurrent = linkCompartments
    (this,compL,time,V,stimulationDesc,iStim,Ve,C)
183
184             global activation;
185             stimulusCurrentIntercomp=zeros(compL,1);
186             stimulusCurrentExtracomp=zeros(compL,1);
187
188             pulsTime1 = stimulationDesc(1,1);           pulseMagnif1 =
    stimulationDesc(1,2);
189             pulsTime2 = stimulationDesc(2,1)+pulsTime1; pulseMagnif2 =
    stimulationDesc(2,2);
190             pulsTime3 = stimulationDesc(3,1)+pulsTime2; pulseMagnif3 =
    stimulationDesc(3,2);
191             pulsTime4 = stimulationDesc(4,1)+pulsTime3; pulseMagnif4 =
    stimulationDesc(4,2);
192
193             %% Apply somatic current, if enabled
194             % The first compartment gets the stimulus from Istim, all other
195             % compartments get the stimulus from their neighbor comp %
196             %
197             % istim->[Soma][][][][][][]
198             %
199             if (iStim~=0)
200                 % stimulusCurrentIntercomp(1)=this.rStim(time,pulsTime1,
    pulsTime2, iStim);
201                 if (time < pulsTime1)
202                     %stimulusCurrentExtracomp = this.compartmentStim
    (this,compL,Ve, pulseMagnif1);
203                     stimulusCurrentIntercomp(1)=pulseMagnif1.*iStim;
204                 elseif (time < pulsTime2)
205                     %stimulusCurrentExtracomp = this.compartmentStim
    (this,compL,Ve, pulseMagnif2);
206                     stimulusCurrentIntercomp(1)=pulseMagnif2.*iStim;
207                 elseif (time < pulsTime3)
208                     %stimulusCurrentExtracomp = this.compartmentStim
    (this,compL,Ve, pulseMagnif3);
209                     stimulusCurrentIntercomp(1)=pulseMagnif3.*iStim;
210                 elseif (time < pulsTime4)
211                     %stimulusCurrentExtracomp = this.compartmentStim
    (this,compL,Ve, pulseMagnif4);
212                     stimulusCurrentIntercomp(1)=pulseMagnif4.*iStim;
213                 elseif (time > pulsTime4)
214                     %stimulusCurrentExtracomp = this.compartmentStim
    (this,compL,Ve, 0);
215                     stimulusCurrentIntercomp(1)=0;
216                 end % if (time < pulsTime1)
217
218             else
219                 stimulusCurrentIntercomp(1)=0;
220                 deltaVfirst=-V(1)+V(2);
221                 lastOhmAx=(this.ohmAx(compL-1)+this.ohmAx(compL))/2;
222                 stimulusCurrentIntercomp(1) = deltaVfirst./(lastOhmAx); %
    mV / kOhm = ?A
223                 stimulusCurrentIntercomp(1) = stimulusCurrentIntercomp(1)/2;
224             end % (iStim~=0)
225
226             %% link the compartments
227             % calculate the current between compartments
228             % [][mV]-[mV][][]
229             if (compL>1)
230                 % middle compartments
231                 deltaV1=V(1:compL-2)-V(2:compL-1);
232                 deltaV2=V(3:compL)-V(2:compL-1); % the delta values, shifted
```

```
    at once.
233                 centerOhmAx=(this.ohmAx(1:compL-2)+this.ohmAx(2:compL-1))/2;
234                 stimulusCurrentIntercomp(2:compL-1) = (deltaV1 + deltaV2)./
    (centerOhmAx.*2); % mV / kOhm
235
236                 %last compartment
237                 deltaVlast=V(compL-1)-V(compL);
238                 lastOhmAx=(this.ohmAx(compL-1)+this.ohmAx(compL))/2;
239                 stimulusCurrentIntercomp(compL) = deltaVlast/(lastOhmAx); %
    mV / kOhm
240
241             end % if (compL>1)
242
243             %% Apply the external activating current
244             % Use the Ve for getting the influence of the external voltage on
    the core of the compartment
245             %
246             % ----Ve---
247             % [ ][Vi][]
248             if (compL>1)
249                 if (time < pulsTime1)
250                     stimulusCurrentExtracomp = this.compartmentStim
    (this,compL,Ve, pulseMagnif1);
251                 elseif (time < pulsTime2)
252                     stimulusCurrentExtracomp = this.compartmentStim
    (this,compL,Ve, pulseMagnif2);
253                 elseif (time < pulsTime3)
254                     stimulusCurrentExtracomp = this.compartmentStim
    (this,compL,Ve, pulseMagnif3);
255                 elseif (time < pulsTime4)
256                     stimulusCurrentExtracomp = this.compartmentStim
    (this,compL,Ve, pulseMagnif4);
257                 elseif (time > pulsTime4)
258                     stimulusCurrentExtracomp = this.compartmentStim
    (this,compL,Ve, 0);
259                 end
260             end % if (compL>1)
261             %stimulusCurrentExtracomp=0;
262
263             %% Sum up the results
264             stimulusCurrent=stimulusCurrentExtracomp+stimulusCurrentIntercomp;
265             if(C~=0)
266                 % some loging
267                 activatingFunction=max(stimulusCurrentExtracomp)/
    C;
268                 activation.activatingFunction=[activation.activatingFunction;
    activatingFunction];
269                 activation.time = [activation.time; time];
270             end
271
272             if (activation.current==0)
273                 activation.current=stimulusCurrent;
274                 if (activation.current~=0) display(['Current measured at ms:
    ' num2str(time) ]); end
275             end % if (activation.current==0)
276         end % function linkCompartments
277
278         function X = surfaceToVolumeRatio(r,l)
279             thisVolume=r.*r*pi()*l;
280             surface=2.*r.*pi().*l+2.*r.*r.*pi();
281             X=surface./thisVolume;
282         end %function
283     end % methods (Static)
284 end % classdef basicModel
```

```
classdef fcn2010_cat_20deg < basicModel
    % Author: Watzinger Anton
    % Last edited: 2017-02-11

    properties
        simulationTime=5; %ms
        t0; timeout=20; %s

        gasConstant=8.31441; % (J/mol*K)
        Faraday=96485.3365; %c/mol
        RTdivF; %mV
        area;
        volume; Temp;
        gNa; gK; gCa;
        T0=35; T=23.5;

        %% Temperature dependent factors
        factor_Q10_Na_kin; factor_Q10_K_kin; factor_Q10_Ca_kin
        factor_Q10_Na_gBar; factor_Q10_K_gBar; factor_Q10_Ca_gBar;

        %% channel densities soma
        gNaSoma=69.4;%(mS/cm^2)
        gKSoma=32;% (mS/cm2)
        gCaSoma=1.39;% (mS/cm2)

        %% channel densities initial segment
        gNaIS=100;%(mS/cm^2)
        gKIS=50.10;% (mS/cm2)
        gCaIS=0.836;% (mS/cm2)

        %% channel densities axon
        gNaAxon=124;%(mS/cm^2)
        gKAxon=50;% (mS/cm2)
        gCaAxon=04;% (mS/cm2)

        gKCa=0.05;% (mS/cm2)
        gL=0.1; % (mS/cm2)


        %% elibrium potentials
        VNa = 60.60; %(mV)
        VK = -101.34; % (mV)
        VL= -65; %(mV)

        %% Calcium properties
        CaRes=10^-7*10^3; % in M/dm3 => mM/cm3
        CaDiss=10^-6*10^3; % in M/dm3 => mM/cm3
        Cao=1.8; % mM

        %% misc properties
        Vrest= -70; % (mV)
        Rn = 150*10^6; %(ohm) = 150 MOhm, input resistance
        r;compartmentLength;
        C = 1; %(mF/cm2)
        TauCA=1.5; %

        %% simulation
        iPause=0.5; % ms
        iStimTime=0.120;
        % would be; [0.5,0.12,0,0; 0,1,0,0]

        % soma stimulation
        iStim=0; %(uA/cm^2) Amount of current stimulation

        % biphasic stimulation: time which the mentioned multiplyer will be
applied %
        % format: (time in [ms], pot multiplicator [1] ; )
        % [ puls1,pot1 ; puls2,pot2  ; puls3,pot3 ; puls4,pot4]
        % exmaple: [0.5,0.15,0.2,0.1; 0,-1,0,+1]
```

```
69
70          % intracellular stimulation:
71          stimulationDescInt=[0.5,0; 0.2,1; 4,0; 0.2,0];
72
73          % extracellular stimulation:
74          % stimulationDescExt=[0.5,0; 0.2,1; 0.5,0; 0.4,-0.5]; % with
    electrode uncharge
75          stimulationDescExt=[0.5,0; 0.2,1; 1,0; 0.4,0]; % without electrode
    uncharge
76
77          startParameters;
78          numberOfStateVariables=5; % h m n c Cai
79          Ve=0;
80          rhoFiber = 1.1; %[Ohm*m]
81          stimulationDesc; ohmAx;
82
83      end %properties
84
85      methods
86
87          %% The standard constructor sets the temperature of the model.
88          function this = fcn2010_cat_20deg(geometryProperties)
89
90              global currentObserver;
91              global temperature;
92
93              this.Temp=273.15+this.T;
94              temperature = this.T;
95
96              %% Temperature dependent factors
97              this.factor_Q10_Na_kin=1.95^((this.T-this.T0)/10);
98              this.factor_Q10_K_kin=1.9^((this.T-this.T0)/10);
99              this.factor_Q10_Ca_kin=1.95^((this.T-this.T0)/10);
100
101             this.factor_Q10_Na_gBar=1.64^((this.T-this.T0)/10);
102             this.factor_Q10_K_gBar=1.49^((this.T-this.T0)/10);
103             this.factor_Q10_Ca_gBar=1.64^((this.T-this.T0)/10);
104
105             %% channel densities soma
106             this.gNaSoma=this.gNaSoma*this.factor_Q10_Na_gBar;%(mS/cm^2)
107             this.gKSoma= this.gKSoma*this.factor_Q10_K_gBar;% (mS/cm2)
108             this.gCaSoma=this.gCaSoma*this.factor_Q10_Ca_gBar;% (mS/cm2)
109
110             %% channel densities initial segment
111             this.gNaIS=this.gNaIS*this.factor_Q10_Na_gBar;%(mS/cm^2)
112             this.gKIS=this.gKIS*this.factor_Q10_K_gBar;% (mS/cm2)
113             this.gCaIS=this.gCaIS*this.factor_Q10_Ca_gBar;% (mS/cm2)
114
115             %% channel densities axon
116             this.gNaAxon=this.gNaAxon*this.factor_Q10_Na_gBar;%(mS/cm^2)
117             this.gKAxon=this.gKAxon*this.factor_Q10_K_gBar;% (mS/cm2)
118             this.gCaAxon=this.gCaAxon*this.factor_Q10_Ca_gBar;% (mS/cm2)
119
120
121             this.Ve=geometryProperties.externalVoltages;
122             this.iStim=geometryProperties.iStim;
123             this.r=geometryProperties.compartmentRadius*100; % m => cm
124
    this.compartmentLength=geometryProperties.distanceBetrweenCompartments*100; %
    m => cm
125             this.area=this.r.^2*pi;   % area in cm2
126             this.volume=(this.compartmentLength)*this.area;
127
128             if (this.iStim==0)
129                 display('using extracellular stimulation function');
130                 this.stimulationDesc=this.stimulationDescExt;
131             else
132                 display('using intercellular stimulation function');
133                 this.stimulationDesc=this.stimulationDescInt;
```

```
134             end % (this.iStim~=0)
135
136             % channel densities vector
137             this.gNa = densityVectorFactory(geometryProperties.compFeatures,
    {'Soma' 'IS' 'SOCB' 'Axon'},
    [this.gNaSoma,this.gNaIS,this.gNaAxon.*10,this.gNaAxon]);
138             this.gK = densityVectorFactory(geometryProperties.compFeatures,
    {'Soma' 'IS' 'SOCB' 'Axon'},[this.gKSoma,this.gKIS,this.gKAxon,this.gKAxon]);
139             this.gCa = densityVectorFactory(geometryProperties.compFeatures,
    {'Soma' 'IS' 'SOCB' 'Axon'},
    [this.gCaSoma,this.gCaIS,this.gCaAxon,this.gCaAxon]);
140
141             % get the number of compartments from the voltage array%
142             numberOfCmp=geometryProperties.numberOfCompartments;
143
144             %% current analysis
145             currentObserver.compartment=1; currentObserver.time = 0;
    currentObserver.INa = 0; currentObserver.IK = 0;
146             currentObserver.IKA = 0; currentObserver.IKCa = 0;
    currentObserver.ICa = 0; currentObserver.IL = 0;
147
148             V0=repmat(this.Vrest,1,numberOfCmp)';
149             this.startParameters=[V0 this.h0(numberOfCmp) this.m0
    (numberOfCmp) this.n0(numberOfCmp) this.c0(numberOfCmp) this.Cai0
    (numberOfCmp)];
150             this.RTdivF=(this.gasConstant*this.Temp)/this.Faraday;
151
152         end
153
154         %% the differential equations
155         function X = differentialEq(this,time,x)
156             compL=length(x)/(this.numberOfStateVariables+1);
157
158             global currentObserver;
159
160             %prevents error if t0 is not initialized
161             if (isempty(this.t0)) this.t0=clock; end
162
163             ms = round(etime(clock,this.t0) * 1000);
164             if(ms>this.timeout*1000)
165                 display(['Timeout, maxPotential:',num2str(max(x(1:compL)))]);
166                 global exceptionValue; exceptionValue=x(1:compL);
167                 return;
168             end
169
170             % slice up the column vector to a 'compL'x6 array
171
172             V=x(1:compL); h=x(compL+1:2*compL); m=x(2*compL+1:3*compL); n=x
    (3*compL+1:4*compL);
173             c=x(4*compL+1:5*compL); Cai=abs(x(5*compL+1:6*compL));
174             thisVe=this.Ve;
175
176             %% get the stimulus power for each compartment, if the neuron
    gets stimulated in first compartment
177             stimulusCurrent=this.linkCompartments
    (this,compL,time,V,this.stimulationDesc,this.iStim, thisVe, this.C); %
    current in mA %
178
179             %% allocate memory
180             X=zeros(compL,6);
181
182             %% Main equation
183             thisINa=zeros(compL,1); thisIK=zeros(compL,1); thisIKCa=zeros
    (compL,1);
184             thisICa=zeros(compL,1); thisIL=zeros(compL,1);
185
186             % to disable single currents, just comment the according line
187             thisINa=this.INa(V,m,h);
188             thisIK=this.IK(V,n);
```

```
189             thisIKCa=this.IKCa(V,Cai);
190             thisICa=this.ICa(V,c,Cai);
191             thisIL=this.IL(V);
192 
193             currentObserver.time = [currentObserver.time; time];
    currentObserver.INa = [currentObserver.INa; thisINa
    (currentObserver.compartment)];
194             currentObserver.IK = [currentObserver.IK; thisIK
    (currentObserver.compartment)];
195             currentObserver.IKCa = [currentObserver.IKCa; thisIKCa
    (currentObserver.compartment)]; currentObserver.ICa = [currentObserver.ICa;
    thisICa(currentObserver.compartment)];
196             currentObserver.IL = [currentObserver.IL; thisIL
    (currentObserver.compartment)];
197 
198             X(:,1)=(-thisINa-thisIK-thisIKCa-thisICa-thisIL+stimulusCurrent)./
    this.C;%Vm [mV];
199 
200             X(:,2)=(-(this.alpha_h(this,V)+this.beta_h(this,V)).*h
    +this.alpha_h(this,V)).*this.factor_Q10_Na_kin; %h
201             X(:,3)=(-(this.alpha_m(this,V)+this.beta_m(this,V)).*m
    +this.alpha_m(this,V)).*this.factor_Q10_Na_kin; %m
202             X(:,4)=(-(this.alpha_n(this,V)+this.beta_n(this,V)).*n
    +this.alpha_n(this,V)).*this.factor_Q10_K_kin; %n
203             X(:,5)=(-(this.alpha_c(this,V)+this.beta_c(this,V)).*c
    +this.alpha_c(this,V)).*this.factor_Q10_Ca_kin; %c
204             X(:,6)=(this.surfaceToVolumeRatio(this.r,this.compartmentLength).*
    (-this.ICa(V,c,Cai)./(this.Faraday*2)) * 10^-3)-((Cai-this.CaRes)./
    this.TauCA);%Cai
205 
206             %create a column vector
207             X=X(:);
208         end
209 
210         %% Helper functions
211         function X=INa(this,V,m,h) % check
212             X=this.gNa.*m.^3.*h.*(V-this.VNa);
213         end %function INa
214 
215         function X=IK(this,V, n)
216             X=this.gK.*n.^4.*(V-this.VK);
217         end %function IK
218 
219         function X=IKCa(this,V,Cai)
220             X=this.gKCaBar(this,Cai) .* (V-this.VL);
221         end %function IKCa
222 
223         function X=ICa(this,V,c,Cai) % check
224             relative_VCa=this.VCa(Cai);
225             X=this.gCa.*c.^3.*(V-relative_VCa);
226         end %function ICa
227 
228         function X=VCa(this, Cai)
229             RTdivF2=(this.gasConstant*this.Temp)/(this.Faraday.*2);
230             X=RTdivF2.*log(this.Cao./Cai); % in volt
231             X=X*10^3; % V => mV
232         end %function VCa
233 
234         % Leak current
235         function X=IL(this,V)
236             X=this.gL *(V-this.VL);
237         end %function IL
238 
239         %% start Values
240         function X = h0(this,l)
241             X=this.alpha_h(this,this.Vrest)./(this.alpha_h(this,this.Vrest)
    +this.beta_h(this,this.Vrest));
242             X=repmat(X,1,l)';
243         end %function
```

```
244
245         function X = m0(this,l)
246             X=this.alpha_m(this,this.Vrest)./(this.alpha_m(this,this.Vrest)
    +this.beta_m(this,this.Vrest));
247             X=repmat(X,1,l)';
248         end %function
249
250         function X = n0(this,l)
251             X=this.alpha_n(this,this.Vrest)./(this.alpha_n(this,this.Vrest)
    +this.beta_n(this,this.Vrest));
252             X=repmat(X,1,l)';
253         end %function
254
255         function X = c0(this,l)
256             X=this.alpha_c(this,this.Vrest)./(this.alpha_c(this,this.Vrest)
    +this.beta_c(this,this.Vrest));
257             X=repmat(X,1,l)';
258         end %function
259
260         function X = Cai0(~,l)
261             X=1.0000e-04;
262             X=repmat(X,1,l)';
263         end %function
264
265     end %methods
266
267     methods (Static)
268         function X = alpha_h(this,V)
269             temperature = this.T;
270             X=0;
271             if (temperature<24)
272                 X=this.factor_Q10_Na_kin*1.869.*exp(-(V+55)/20);
273             elseif(temperature>30)
274                 X=this.factor_Q10_Na_kin*1.817.*exp(-(V+52)/20);
275             end
276         end %function
277
278         function X = beta_h(this,V)
279             temperature = this.T;
280             X=0;
281             if (temperature<24)
282                 X=this.factor_Q10_Na_kin*28.04./(1+exp(-0.1*(V+25)));
283             elseif(temperature>30)
284                 X=this.factor_Q10_Na_kin*27.25./(1+exp(-0.1*(V+22)));
285             end
286         end %function
287
288         function X = alpha_m(this,V)
289             temperature = this.T;
290             X=0;
291             if (temperature<24)
292                 X=(this.factor_Q10_Na_kin*-2.804*(V+35))./(exp(-0.1*(35
    +V))-1);
293             elseif(temperature>30)
294                 X=(this.factor_Q10_Na_kin*-2.725*(V+35))./(exp(-0.1*(35
    +V))-1);
295             end
296         end %function
297
298         function X = beta_m(this,V)
299             temperature = this.T;
300             X=0;
301             if (temperature<24)
302                 X=this.factor_Q10_Na_kin*93.46*exp(-(V+60)/18);
303             elseif(temperature>30)
304                 X=this.factor_Q10_Na_kin*90.83*exp(-(V+60)/20);
305             end
306         end %function
307
```

```
308          function X = alpha_n(this,V)
309              temperature = this.T;
310              X=0;
311              if (temperature<24)
312                  X=(this.factor_Q10_K_kin*-0.0984*(V+32.5))./(exp(-0.1*(V
     +32.5))-1);
313              elseif(temperature>30)
314                  X=(this.this.factor_Q10_K_kin*-0.09575*(V+37))./(exp(-0.1*(V
     +37))-1);
315              end
316          end %function
317
318          function X = beta_n(this,V)
319              temperature = this.T;
320              X=0;
321              if (temperature<24)
322                  X=this.factor_Q10_K_kin*1.969*exp(-(V+58.5)/76);
323              elseif(temperature>30)
324                  X=this.factor_Q10_K_kin*1.915*exp(-(V+47)/80);
325              end
326          end %function
327
328          function X = alpha_c(this,V)
329              temperature = this.T;
330              X=0;
331              if (temperature<24)
332                  X=(this.factor_Q10_Ca_kin*-1.4*(V+15))./(exp(-0.1*(V+15))-1);
333              elseif(temperature>30)
334                  X=(this.factor_Q10_Ca_kin*-1.362*(V+13))./(exp(-0.1*(V
     +13))-1);
335              end
336          end %function
337
338          function X = beta_c(this,V)
339              temperature = this.T;
340              X=0;
341              if (temperature<24)
342                  X=this.factor_Q10_Ca_kin*46.68*exp(-(V+40)/18);
343              elseif(temperature>30)
344                  X=this.factor_Q10_Ca_kin*45.41*exp(-(V+38)/18);
345              end
346          end %function beta_c
347
348          function X = gKCaBar(this,Ca2i)
349              X = this.gKCa*((Ca2i./this.CaDiss).^2)./(1+(Ca2i/this.CaDiss).^2);
350          end % function gKCaBar
351
352          function [T, Y] = postProcessing(this,T,Y)
353              h.figure = figure;
354              plotaxis.Xmin=0; plotaxis.Xmax=this.simulationTime;
     plotaxis.Ymin=-90; plotaxis.Ymax=50;
355              plot(T,Y(:,1,1));
356              axis([plotaxis.Xmin plotaxis.Xmax plotaxis.Ymin plotaxis.Ymax]);
357
358              filename=[  '/tmp/membranPotential/'  'fcn2010_cat20' '+' num2str
     (this.stimulationDesc(3,1)) 'ms' '+0uA'];
359              filename=strrep(filename, '.', '_');
360
361              if ~isequal(exist('/tmp/membranPotential', 'dir'),7) % 7 =
     directory.
362                  mkdir('/tmp/membranPotential');
363              end %  if ~isequal(exist('/tmp/membranPotential', 'dir'),7)
364
365              close all;
366          end %function postProcessing
367
368      end %method static
369  end %classdef
```

A.2 Ca-Kinetics Correction factor

Basic equation for i_{Ca} dependant Ca-Kinetics, equal to Equation 2.26:

$$\frac{d[Ca^{2+}]_i}{dt} = \frac{A}{V} \cdot \frac{-i_{Ca}}{2F} - \frac{[Ca^{2+}]_i - [Ca^{2+}]_{res}}{\tau_{Ca}} \tag{A.1}$$

$$\frac{d[Ca^{2+}]_i}{dt} = \frac{A}{V} \cdot -i_{Ca} \cdot \frac{1}{2F} - \frac{[Ca^{2+}]_i - [Ca^{2+}]_{res}}{\tau_{Ca}} \tag{A.2}$$

A conversion factor x is needed to fit the different input units into the Ca-Kinetics equation. The equation is solved for x which represents the correction factor:

$$\frac{d[Ca^{2+}]_i[mM]}{dt[ms]} = x \cdot \frac{A[cm^2]}{V[cm^3]} \cdot \frac{-i_{Ca}[\mu A]}{[cm^2]} \cdot \frac{1}{2F[C \cdot mol^{-1}]} - \frac{[Ca^{2+}]_i[mM] - [Ca^{2+}]_{res}[mM]}{\tau_{Ca}[ms]} \tag{A.3}$$

Defined Conversions:

$$M = \frac{mol}{l} \tag{A.4}$$

$$C = A \cdot s \tag{A.5}$$

Values removed, only units used to evaluate the unit-correction factor:

$$\frac{[mM]}{[ms]} = x \cdot \frac{1}{[cm]} \cdot \frac{[\mu A]}{[cm^2]} \cdot \frac{1}{[C \cdot mol^{-1}]} - \frac{[mM] - [mM]}{[ms]} \tag{A.6}$$

$$\frac{[mM]}{[ms]} = x \cdot \frac{1}{[cm]} \cdot \frac{10^{-6}[A]}{[cm^2]} \cdot \frac{[mol]}{[C \cdot]} - \frac{[mM]}{[ms]} \tag{A.7}$$

$$\frac{[mM]}{[ms]} = x \cdot \frac{1}{[cm^3]} \cdot 10^{-6}[A] \cdot \frac{[M \cdot l]}{[C]} - \frac{[mM] - [mM]}{[ms]} \tag{A.8}$$

$$\frac{[mM]}{[ms]} = x \cdot \frac{1}{[cm^3]} \cdot 10^{-6}[A] \cdot \frac{[M \cdot l]}{[A \cdot s]} - \frac{[mM]}{[ms]} \tag{A.9}$$

$$\frac{[mM]}{[ms]} = x \cdot \frac{1}{[cm^3]} \cdot 10^{-6}[A] \cdot \frac{[M \cdot dm^3]}{[A \cdot s]} - \frac{[mM]}{[ms]} \tag{A.10}$$

$$\frac{[mM]}{[ms]} = x \cdot \frac{1}{[cm^3]} \cdot 10^{-6}[A] \cdot \frac{[M \cdot cm^3 \cdot 10^3]}{[A \cdot s]} - \frac{[mM]}{[ms]} \tag{A.11}$$

$$\frac{[mM]}{[ms]} = x \cdot 10^{-6} \cdot \frac{[M \cdot 10^3]}{[s]} - \frac{[mM]}{[ms]} \tag{A.12}$$

$$\frac{[mM]}{[ms]} = x \cdot 10^{-3} \cdot \frac{[M]}{[s]} - \frac{[mM]}{[ms]} \tag{A.13}$$

$$\frac{[mM]}{[ms]} = x \cdot 10^{-3} \cdot \frac{[mM]}{[ms]} - \frac{[mM]}{[ms]} \tag{A.14}$$

$$10^3 = x \tag{A.15}$$

Appendix B

FEM Mesh

B.1 FEM Mesh of Bulk Volume

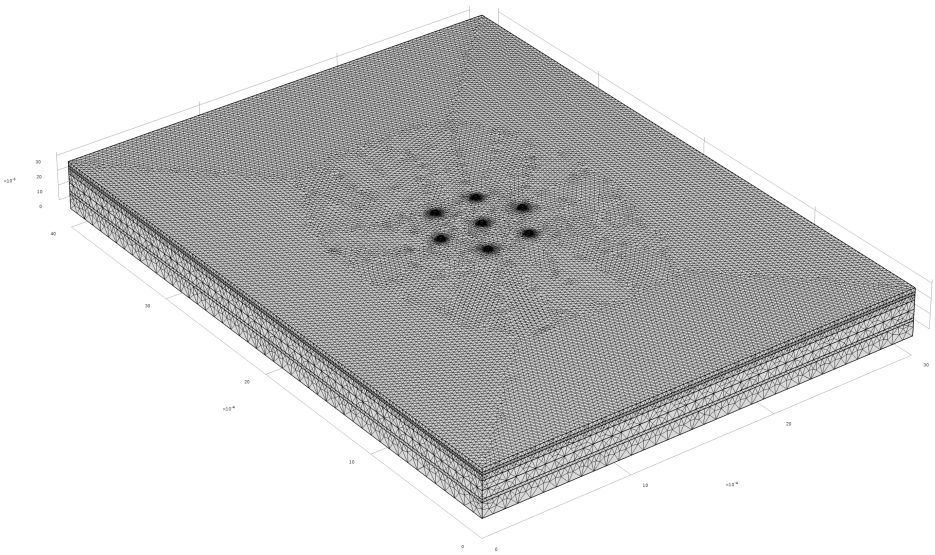


Figure B.1: Complete mesh of the volume conductor consists of 604213 domain elements, 204264 boundary elements, and 3570 edge elements. Exported from Comsol Multiphysics.