RWTH Aachen University
Software Engineering Group

First line of title
Second line of title
Third line of title
Forth line of title

Bachelor Thesis/Master Thesis/Seminar Paper

presented by

Surname, Prename

1st Examiner: Prof. Dr. B. Rumpe

2nd Examiner:

Advisor:

The present work was submitted to the Chair of Software Engineering

Aachen, January 10, 2022

Kurzfassung

Eine kurze Zusammenfassung der Arbeit.

Abstract

A short abstract of this thesis.

Contents

Chapter 1

Introduction

-Neuropathic pain as basis
-comes with many diseases

-pain as electric signals
-goal to understand the firing patterns in nerve fibers
-microneurography as recording technique
-needle in vitro in patients
-action potentials as spikes
-animal data
-does not need fiber separation

-OpenMNGlab
-currently only good for loading the data
-want to add analysis capabilities
-compute quantifiers for spike trains and recordings
-discuss results

Software:
-working on jupyter notebook
-automate the spike analysis process
-integrate analysis into openMNGlab
-get requirements for spike analysis software

There are many unsolved problems in the medical sciences. One of which is neuropathic pain that often occurs as a side effect of other diseases such as Diabetes for example.
-TODO: Do more research on neuropathic pain
Pain is transmitted as electric signals through the nerves inside our bodies. Because of this fact in order to understand different types of pain such as neuropathic pain for one we need to understand the transmission of electrical signals in nerve fibers.
Measuring electrical signals can be done in different ways. One of which is called microneurography. This is a technique that is used to record nerve activity in peripheral nerves. With this technique typically a needle gets inserted into a nerve fiber which then detects the electrical current in the fiber. Additionally, we can stimulate the nerve fiber to get certain responses.
Nerve fibers transmit data with the use of action potentials, short AP or spike. It has been

shown in previous research that information is not transmitted by the shape or amplitude of the spikes, but the frequency and timings (look the exact papers up again).

[**?**]

Chapter 2

Data

Because human nerve data is hard to obtain, we can also use animal data instead as a proxy. Animal data is usable as proxy because we can observe the nerve fibers in vitro but can better separate one single nerve fiber from others. In human data an additional step of fiber separation is necessary to differentiate between individual fibers. We can use the same experimental protocols on Animals as we would on humans. This way we can understand firing patterns of spikes and quantify them. The results can then be applied to human data.
In the case of this thesis, we are using the data from wistar rats. The data was recorded from 2011 to 2012 by Roberto de Col and was published in a paper (put reference). The goal of the paper was to evaluate the effects of spiking activity on the response to mechanical stimulation.

-How much of the exact experimental details are supposed to go here as far as methodology goes, since this is a computer science thesis

The experiments were done in vitro on peripheral nerve fibers. The fibers were mechanically and electrically stimulated via a custom made electromechanostimulator. The nerve activity was recorded using an electrode. The electrical stimulation consists of small electrical pulses that come in a controlled frequency. The mechanical force is applied in a sinusoidal shape.
For single recordings the mechanical force that is applied throughout stays at approximately the same level for most of the files (put for how many files this is the case), but there are exceptions where the mechanical stimulation changes in amplitude and length during one recording.

The experimental software used for these experiments is called Spike2 and is described further in the background chapter.

Chapter 3

Background

3.1 Analysis tools

When it comes to analysis software for microneurography data, there are multiple available that each have their own use-cases.

3.1.1 FieldTrip

Fieldtrip (fieldtriptoolboX.org) is a MATLAB toolbox developed at the Radboud University, Nijmegen, the Netherlands and offers a wide variety of analysis functions. It can analyze MEG, EEG, and iEEG and is an open-source software that has been in development since 2003. Its main strengths lie in the analysis of noninvasive and invasive electrophysiological data. It provides over 100 high-level and 800 low-level functions that can be used both by experimental users as well as developers. It does not feature a graphical user interface, but instead focused on providing direct access to the high and low-level analysis functions that can be used in the command line or in scripts. This is done because FieldTrip is not meant to be an application but a set of tools that can be mixed and matched for different requirements. With this approach the user either needs to be somewhat familiar with MATLAB before starting their analysis, but after a potential initial time invest the flexibility of this approach yields great advantages. The analysis functions are meant to be combined and used as a sort of analysis protocol, where the output of functions is used to compute the next results.

This software package is very useful and widely used in the field, however it does not lend itself to our specific needs. The main problem with this software is its programming language. It is a MATLAB toolbox, however it would be preferred to use a software package in python or another programming language that slots better in the already existing structure that is used at the chair for medical informatics.

3.1.2 Elephant

Elephant (https://elephant.readthedocs.io/en/latest/index.html) is a python module which offers some high-level analysis functions for spike trains specifically. The main problem with this software is the lack of basic functionalities. It relies more on highly specified analysis tools that are not necessarily viable in our use-case. For the use in this thesis I want to start with the basic signal from the spikes, try out different quantifiers and look at the data from a fresh perspective.

3.1.3 openMNGlab

These are two analysis tools I presented that could not be used for this thesis because of different reasons. There are multiple data acquisition software solutions that are used at the chair of IMI that each produce different file formats.. The three main acqusition software packages that need to be handled are Spike2, Dapsys and OpenEphys and are described in more detail in the next section.
We need a tool that that has the capability to load files from these different data acqusition softwares, put them in a compatible format and analyze them further. FieldTrip does not offer these kinds of capabilities as well as it being a matlab package, which would not be ideal for fitting into the rest of the analysis systems at IMI. Elephant is also lacking the importing tools that would be required for the software solution.
For this reason the Institute of medical informatics has started to develop their own software framework in python called openMNGlab. This framework aims to provide a solution for those issues with the different file types and combine them into a single usable format. In addition it provides analysis capabilities for microeurography data.
For the import of the different file types from the data acqusition packages and the internal structure of the data openMNGlab makes use of an already developed python module called Neo(neuralensemble.org/neo). This is a package that already deals with neurophysiological data in various different file formats. It is aimed towards combining these different sources while not providing any analysis capabilities of its own to reduce dependencies (cite website here). It also comes with templates to build importers for different new file formats should those be required and thus makes it easier to include more file formats at a later date.
OpenMNGlab has started as a project of the Institute of medical informatics and was initially developed by Fabian Schlebusch who developed openMNglab 1.0 and setup the basic structure and developed the importers necessary for the file formats that we require. However, the 1.0 version of the framework is not enough in terms of analysis and also importing capabilities and needs to be improved.

At IMI there are people who work on varying aspects of MNG data analysis such as analyzing the response latency to electrical stimulation or analyzing the nerve activity in response to certain chemicals. I am focusing on analyzing spike trains occuring in mechanically and electrically stimulated nerve fibers.
With my bachelor thesis I aim to provide some more analysis functionalities for spike trains as well as give some ideas on the software engineering of the software framework in the future.
In the end FieldTrip and Elephant offer good solution for slightly different use-cases, but were not a perfect fit for the analysis of microneurography data at the Institute of medical informatics. They are, however, a good inspiration for what should be included in such a software framework. In the case of Elephant, since it is also a python module and also makes use of the Neo structure, it could make for a good addition of more high-level analysis functionalities in the future.

Neo:
brings a data structure

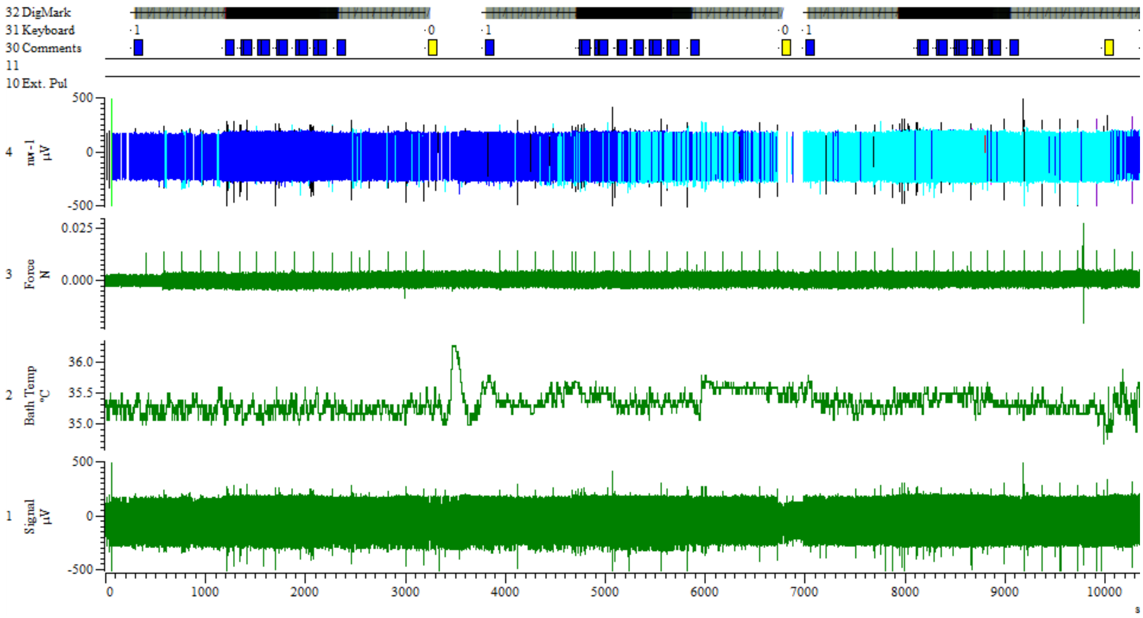

Figure 3.1: Typical mechanically and electrically stimulated recording in Spike2

importers or at least templates for importers

3.2 Data acquisition software

A data acquisition system is a combination of software and hardware components that work together in order to control inputs towards and record data from different subjects. There are many different data acquisition systems for electrophysiological data. There are three systems that we are working with and that should be compatible with our chosen software analysis framework. These systems are Spike2, Dapsys and OpenEphys.

3.2.1 Spike2

Spike2 is a data acquisition and analysis software produced by Cambridge electronic design limited. It is a flexible tool that can be used in a variety of different ways.

The software can record multiple channels simultaneously. An example screenshot from a recording can be seen in Figure 3.1. This depicts a typical recording used for analysis in this bachelor thesis. The recording contains data from nerve fibers of rat cranial dura mater. The nerve fibers were stimulated using a mechanoelectrostimulator applying electaical and mechanical stimulation.

First of all it contains a channel for the recorded raw signal at the bottom. The next channel contains the temperature during the recording. In this example it fluctuates between 35°C and 36.5°C. In channel 3 we can observe the mechanical force that was applied to the nerve fibers. In Figure 3.1 there are spikes in mechanical force whenever a mechanical stimulation occurs to evoke a spike train. For this experiment we want to collect the data of single nerve fibers. It is diffucult, to record just a single nerve fiber in vitro, however. This is why in this experiment spike templates are applied to the raw signal

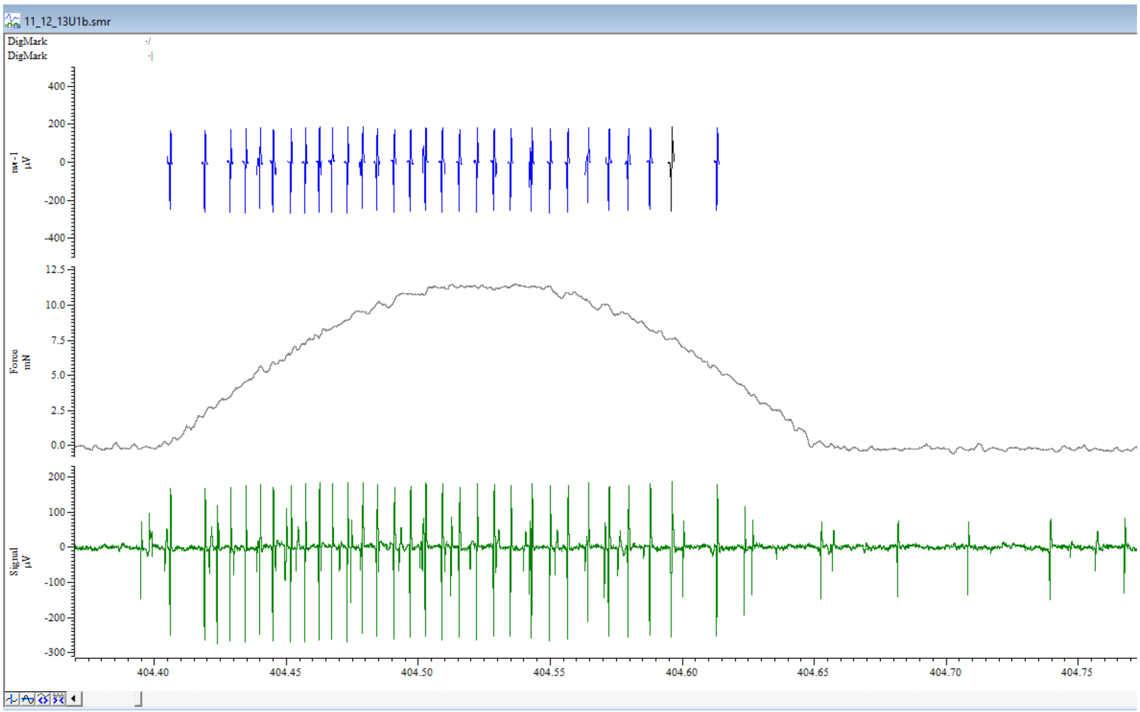

Figure 3.2: A single spike train in spike2

to filter out specific fibers. These filtered fibers are then displayed in so called wavemark channels. In this example channel 4 is such a channel, where only specific action potentials are filtered. The filtering process is done by the experimenters and is based on certain features of the action potential shape.
The topmost channel in Figure 3.1 contains markers for the electrical and mechanical stimuli. Additionally there is a channel containing comments regarding the experiment. Comments can represent the experimental protocol and are filled in by the experimenters. In this example there are comments denoting a change electrical stimulation frequency. In other experiments for example, these could also denote the application of certain chemicals towards the recorded subject.

A more detailed view of a single spike train can be seen in Figure 3.2. Here the difference in electrical and mechanical event markers in the topmost channel can be seen. Mechanical markers are represented by a slash, while electrical markers are represented by a vertical line. Another thing that can be seen here is the channel containing only the spikes. This channel is ideal for the extraction of the spikes for later analysis as there is no noise in the channel anymore and the spikes can also be interpreted as simple events with a timestamp.

3.2.2 Dapsys

The second data acquisiton package that openMNGlab needs to support is called Dapsys. It is a hardware and software system that can record and analyze electrophysiological data from animal or human sources and has been mainly used for studying the peripheral nervous system (cite website here). It has been in development for over 30 years since its earliest version.
Dapsys offers the capabilities of path tracking and comes with the benefit of much data

being available from experiments conducted with the Dapsys software. It is used especially for mng-experiments with human patients. As is the case for Spike2 it also comes with a visual representation of the data which can be seen in Figure (put ref to screenshot here). This graphical interface works real time while recording the data.
The version of Dapsys used in the experiments from Barbara Namer is specialized for microneurography and was configured in cooperation with Brian Turnquist.
The data can also be imported by openMNGlab, however, unlike Spike2, where we can directly import the experimental file, we first need to do an extra step. We need to export the raw data and templates for the tracks we want to analyze as csv files from the Dapsys software.

3.2.3 OpenEphys

The last data acquisition system I want to focus on is called OpenEphys. It is an open-source electrophysiology data acquisition system originally developed by a nonprofit based in Cambridge, Massachusetts. It sets a big focus on modularity and flexibility so that it can fit many different needs from a variety of users.

The idea behind OpenEphys came after an increased popularity of closed-loop experiments in neuroscience, in which the results of the recorded system has an influence on the system itself. With proprietary systems it is somewhat difficult to share the details of such experiments and to replicate them. The introduction of OpenEphys, an open-source system is supposed to make it easier to develop and share analysis and details of such closed-loop experiments.
OpenEphys makes use of inexpensive open-source hardware such as Intan chips to make it easier for small labs to get started with analysing electrophysiological data. The heart of the software part of the system is a plugin-based graphical user interface (GUI). Here the user can add different modules to the processing pipeline such as modules for communicating with the hardware or modules for analyzing the data.
Different labs often have widely varying sets of requirements towards data analysis and therefore data acquisition systems. At the start of a project they are confronted with the question of whether to use a proprietary system or develop their own systems, which will result in much more effort before the real work can begin. Proprietary systems also often come with very limited customization possibilities, which makes it harder to get the system that fits the users needs exactly. This is where OpenEphys comes in with its plugin-based software. It is designed so that everyone can mix and match the processing modules and end up with the exact analysis pipeline that they require. In addition to a wide array of already available processing modules, it is possible to develop your own modules that fit seamlessly into the structure of the system. OpenEphys is completely developed in C++ and based on a library for audio applications. This makes it easy to develop new modules by making use of the class inheritance capabilities of C++.
OpenEphys comes with many advantages such as low cost, transparency and flexibility due to the modularity, but there are also drawbacks to using the system. The start is also a little harder than in proprietary systems, where there is often a straight forward way of getting started with your first experiments. The modular nature of the system might also effect the performance of the analysis, which is why commercial systems might be a better choice, if they fit the analysis needs.

In this thesis we are considering OpenEphys because it is used by a collaborating research team in Bristol, which also wants to make use of openMNGlab.

Chapter 4

Software

-Use-cases:
  -opening data (importing)
  -latency study (Alina)
  -chemical data study (Jessica)
  -mechanically evoked spike trains (Alexander)
  -experimental researchers (Barbara)
-results in a list of requirements
-Use-cases lead to software engineering approach
-nessecary steps for my analysis
-how I implemented it

The first user is a student who does data analysis on mechanically and electrically stimulated Spike2 data. The goal is to perform latency analysis for spikes. The raw spike2 files feature a lot of information, not all of which is always needed. In this case, all that is required to analyse the latencies of the spikes are the timestamps of the spikes itself and the timestamps of the stimulation events. For the Spike2 software this means that we need to extract the DigMark channel which contains the event information as well as the wavemark channel which contains the information on the already presorted spikes.
The relevant information can be imported using the importing function of openMNGlab.

The second user is also a student who uses mechanically and electrically stimulated Spike2 data. His goal is to analyse spike trains resulting from mechanical stimulation. For this he needs also does not need all of the information contained in the raw Spike2 file. He needs the event information for mechanical and electrical stimuli as well as the mechanical force information for details of the mechanical stimuli. As well as the first user he also needs the information collected from the wavemark channels detailing the exact spiking patterns.

Which information should available after importing data?
Spikes + timestamps
Electrical stimulation + timestamps
Mechanical stimulation + timestamps, duration, amplitude
Information about application of other stimuli (chemicals, heat...)
For human data: temperature?

Spike2
Spike channels + some way to group them easily (e.g., in groups from spike2 templates)

Electrical event channel + some way to distinguish between electrical and mechanical events
Mechanical force channel (maybe optional)
Comments channel (maybe optional)
Temperature (optional) (probably needed for human data)

Chapter 5

Conclusion

Figure 5.1: Das SE Logo

Appendix A

z. B. Programmdokumentation