UNIVERSITÄT ZU LÜBECK
INSTITUTE FOR ELECTRICAL
ENGINEERING IN MEDICINE

A Novel Algorithm for
Electrical Engineering in Medicine

Ein neuer Algorithmus für die
Medizinische Elektrotechnik

Masterarbeit
im Rahmen des Studiengangs Informatik der Universität zu Lübeck

Vorgelegt von
Maxima Musterfrau

Ausgegeben und betreut von
Prof. Dr. Awesome Scientist

Mit Unterstützung von
Lucky PhD Student

Diese Arbeit ist im Rahmen von Arbeiten
bei der Firma PerfectSolutions GmbH entstanden.

Lübeck, den 1. April 2020

Eidesstattliche Erklärung

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne die Benutzung anderer als der angegebenen Hilfsmittel selbstständig verfasst habe; die aus anderen Quellen direkt oder indirekt übernommenen Daten und Konzepte sind unter Angabe des Literaturzitats gekennzeichnet.

Lübeck, den 1. April 2020

Abstract

This is the place for a very short abstract about your work. It should offer the reader an overview about the scope of the work and the attained results – after reading this, a reader should be able to judge whether reading this thesis might prove beneficial to him. So take care about its length and comprehensibility. The abstract for a thesis should span between a half and one page. For the present (shorter) document, an abstract could look like this:

This work gives a short introduction to the typesetting tool LaTeX and points out its advantages for writing a scientific thesis. Additionally, several more general hints on how to write a bachelor's thesis, master's thesis or project work are given, concerning structure, contents and representation.

Kurzzusammenfassung

An diese Stelle kommt eine Kurzzusammenfassung der Arbeit. Dies sollte eine Übersetzung des englischen Abstracts sein.

Contents

1 Introduction

You've just started your project work, your bachelor or master's thesis. Maybe you've already achieved some results like experiments, algorithms, methods or you've made some handwritten notes. Now you wonder how to put that on paper? This document is both a manual on how to write a thesis as well as a template that you can use for your own thesis. Once you've read this document and checked out the accompanying LaTeX-files from which it is generated, you should be able to produce your own scientific work, in a professional and visually appealing form. If you have any feedback (positive or negative) regarding this document, please let me know!

Eike Petersen, March 2019
(eike.petersen@uni-luebeck.de)

FiXme Note: This is a FixMe Note that tells you to complete the introduction of your thesis. To see how this is achieved, see `introduction.tex`. You can use FixMe to help you organizing your thesis writing process, if you like. For more information see the FiXme manual [5].

1.1 Previous Works

This section should contain a quick overview of the previous results of other researchers or students relevant to your thesis project. It serves to clearly position the results of your thesis in relation to other works, and to clarify the novelty of your work in comparison to previous approaches. This is relevant because it communicates to the reader why your work is valuable and useful to the scientific community!

Normally, this section should span between a half and one page, and you should provide many references here. Note that this is not an easy section to write, because it requires you to have a solid understanding of the precise differences between various existing approaches, and how your method relates to them. To be able to do this, you should spend some time searching for, reading, and understanding papers and textbooks relevant to your thesis.

1.2 Thesis outline

This work is structured in two sections. Chapter 2 describes how to produce a LaTeX-file using some illustrative examples. Chapter 3 gives a short introduction to scientific writing techniques.

2 Materials and Methods

This chapter deals with background information relevant for your thesis, including physiological background, existing research on the topic, and mathematical and other preliminaries required to understand the novel concepts presented in the following chapter. It is also often called *Fundamentals* or *Background*.

2.1 Writing the background chapter

As stated above, this chapter should contain the relevant background information that is required to follow the developments and discussions in the following (main) chapters. Once you're beginning to write your thesis, you have probably learned a lot about your subject, and hence it will feel good to write down all that newly-gained knowledge in the background chapter. It is therefore important to stress that you should *not* use this chapter to write down all the background knowledge you have about the subject of your thesis. This is *not* intended to be a comprehensive textbook-style chapter that explains everything there is to know about the topic of the thesis. Instead, really try to focus on *just* the information which is necessary to understand the following chapters. As a rule of thumb, the background chapter should generally not exceed 15 pages.

As an important corollary, it follows that *writing the background chapter first is a bad idea*, because if you have not yet written the other chapters, how can you know which information you need to provide in the background chapter? You should – at the very least – know the detailed structure for your *whole* thesis precisely before beginning to write the background chapter. Otherwise, you will probably find yourself rewriting or even removing large parts of this chapter later on. For more details on the structure of the thesis writing process, refer to chapter 3.

2.2 LaTeX's features

This section provides a very brief introduction to writing a thesis with LaTeX. It is important to know that LaTeX is not a WYSIWYG (what you see is what you get) program like other text editors, such as Microsoft Word. Instead, it much more resembles a programming language, in which you construct your text by proper usage of syntax. The "source code" is your LaTeX file (`.tex`).

As in other programming languages, it is possible to insert comments in LaTeX that are not visible in the final text, using the % sign (refer to the source code for this paragraph in `materials_and_methods.tex` to see an example). Line breaks are not of importance while writing the text, since they are created automatically during compilation. Paragraph breaks are inserted automatically wherever there is a blank line in the source file. The general formatting of the whole document can be customized precisely, but for the beginning, the default formatting provided in this template should be sufficient.

One of the big advantages of LaTeX is its strong support for typesetting mathematical formulae, see section 2.6. The print quality achieved with LaTeX formulae is hardly matched by any other program, let alone free of charge. Also, references to your equations using the correct equation numbers can easily be used and maintained even while you change your document, insert new formulae that change the numbers of all equations, and so on. Citing other works and providing a nicely compiled list of references is very easy in LaTeXas well, see section 2.5.

The following sections are not very useful if you just look at them in their compiled (pdf) form, but if you read the source code at the same time, it is easy to understand how different elements are constructed. You should try to compile the file yourself and compare the results. If there are any differences, check if your LaTeX configuration is correct.

2.3 Getting started with Latex

To get started with Latex, you need...

1. A tool that generates a PDF file out of a bunch of *.tex and *.bib files. On Windows, this is typically `MikTex` (`http://miktex.org/download`).

2. A text editor. This can be as simple as `Notepad++` (`https://notepad-plus-plus.org/`), but many would recommend an IDE that provides further convenience features. `TexMaker` (`http://www.heise.de/download/texmaker.html`) and `TexStudio` (`http://www.texstudio.org/`) are two well-known examples.

If you have these two installed on your PC, you're ready to go! There is a wealth of references and tutorials on the internet that deal with Latex. What follows is a small list compiled based on my personal preferences.

- ShareLatex currently provides - to my taste - the best introduction and reference on a number of Latex-related topics: `https://www.sharelatex.com/learn`.
- Latex-Wikibooks often prove useful if you're really looking for a reference of the available symbols, e.g. `https://en.wikibooks.org/wiki/LaTeX/Mathematics` or `https://en.wikibooks.org/wiki/LaTeX/Tables`.
- There is a neat little online tool available which provides users with hints on available Latex commands based on the user's drawing of the desired symbol: `http://detexify.kirelabs.org/classify.html`.
- Malte Schmitz from the Universität Lübeck also provides good introductory material in German: `http://www.mlte.de/layout`.

2.4 Figures in Latex

Figures should represent an essential component of your thesis, as they are excellent tools for effectively communicating information on complex issues to your reader.

2.4.1 Creating good scientific figures

While it is not easy to create visually appealing figures that represent information in the most accessible way, this is a skill that can be learned over time by reading various tutorials on the topic, carefully examining the works of other authors, and always reviewing one's own figures critically in this regard. A good figure should be both functional by effectively conveying the relevant information, and visually appealing to the reader. In *very* few cases, a standard figure generated using, e.g., Matlab or python should be included directly in a scientific document without further modification and „tuning“.

To achieve functionality, the creator of a figure should think hard about the information which the figure should convey to the reader, and the best way to depict this visually. No unnecessary details should be present in the figure, because they will distract the reader from the main content – grid lines, for example, are not really necessary in many plots and represent a distraction. Every element of the figure that is not completely self-explanatory should be labeled. The colors of elements should be selected in a meaningful way, i.e., very different elements should have clearly differing colors (blue and red, for example), while similar elements should have colors indicating that they belong together. For example, if two different versions of an algorithm are compared to a third, very different algorithm, one might choose two similar (but still clearly distinguishable) colors for the first two, and a very different color for the third one. See `www.colorbrewer2.org` for a list of color maps for many different use cases. (There are packages available for many different programming languages, including python and Matlab, which implement the colorbrewer2 color maps.)

To make the figure visually appealing, care should be taken to make the figure neither too large (wasting space and making the figure look disproportionate in relation to its surroundings) nor too small (obscuring important details), and to pick a color map that looks pleasing. Screen shots and bitmap graphic files should generally be avoided, because they often look bad when scaled, have the wrong dimensions, etc. Text size in figures should be a little bit smaller than normal text size, but still easily readable. Finally, it is very important to ensure consistent formatting between figures in the thesis: ideally, they should all use the same color code, text style and size, and generally have the same „look and feel“.

A minimal checklist regarding the preparation of good scientific figures can be found in the *Scientific Writing Checklist* that comes with this document. A very brief introduction to clean and clear figure design can be found in, e.g., `https://youtu.be/UuL6wPGTJZQ`. For more details on how to create good scientific figures, refer to, e.g., Few [4], Rougier, Droettboom, and Bourne [13], Tufte [14], and Ware [15]. In the following, the rather technical aspect of how to generate and include figures in Latex will be considered, and several options will be discussed.

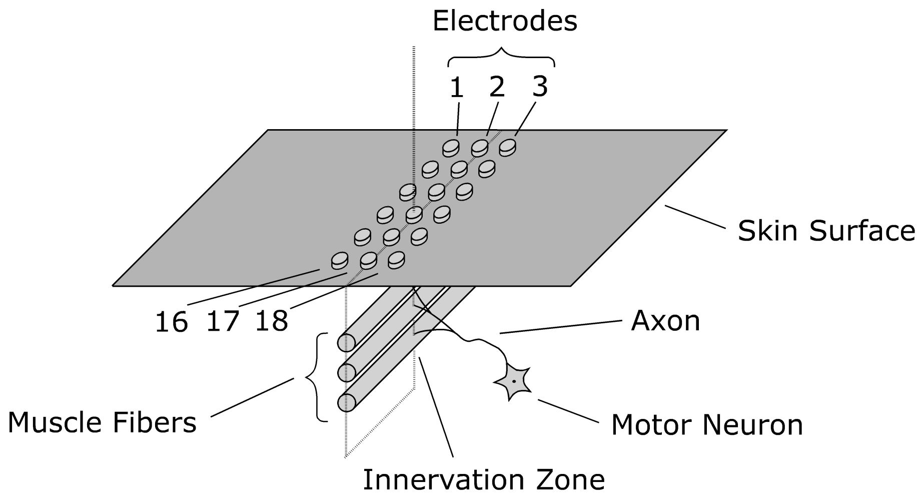


Figure 2.1: Schematic of the geometry in single fiber action potential (SFAP) measurements of human muscle fibers. Shown are three muscle fibers at different depths in the muscle tissue, and a regular grid of 18 recording electrodes on the skin surface. This figure has been produced using the `Inkscape` software.

2.4.2 Including figures in a Latex document

Figure 2.1 shows a schematic of the geometry of of single fiber action potential (SFAP) detection. Note the extensive figure caption: many readers will first skim through your thesis and look at the figures, which should hence be as self-explanatory as possible.

As opposed to fig. 2.1, which is included here as an external graphics file, fig. 2.2 shows a simple electrical schematic that is created completely in Latex, using the `circuitikz` package. This package is based on the powerful `tikz` package which allows for drawing any kind of figure directly in Latex. This bears a number of advantages:

- The resulting figures are vector graphics, i.e., they are small (hence not resulting in a final thesis pdf size of several MBs) and generally look good.

- Font style and size is consistent with the rest of the document. This is a major problem when importing figures from other software, e.g., `Matlab`. (Also compare fig. 2.1.)

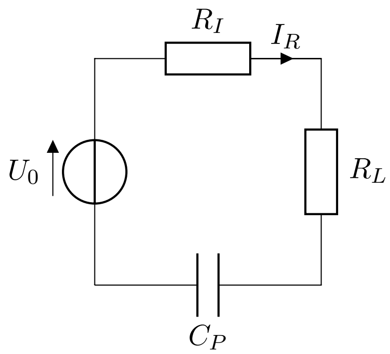


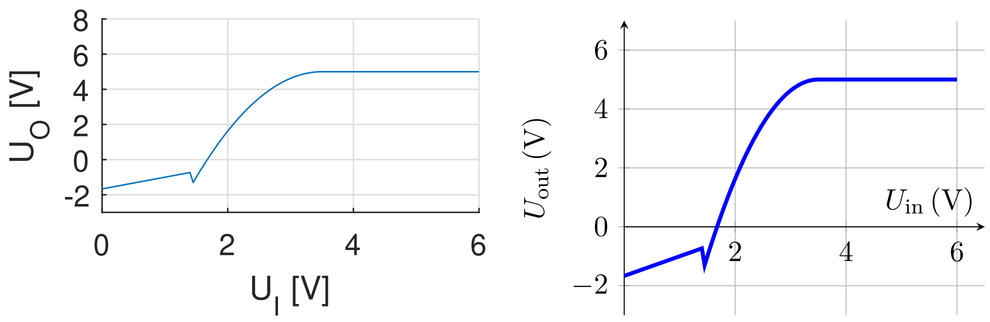
Figure 2.2: A simple electrical circuit, drawn in Latex using the `circuitikz` package. Note how the font type and size are exactly the same as in the normal text body.

- Each figure created this way is completely reproducible and configurable in every aspect.

Another package that is also based on the `tikz` package is the `pgfplots` package which enables the user to create plots either completely in Latex, or import data[1] from an external program (e.g., `Matlab` or `Python`) and generate a nice-looking plot from this data using `tikz` (with the advantages mentioned above). Figure 2.3 shows an exemplary comparison of a plot exported directly from `Matlab` and the same data points exported to a data file and plotted using `pgfplots`. (It also shows how to create multiple subfigures inside a single figure.) Figure 2.4 shows an example of a plot that is generated completely in `pgfplots`, without any aid from an external program.

The `pgfplots` package has one drawback, which is at the same time its greatest advantage: it generates *vector graphics*. Vector graphics have a number of significant advantages over raster graphics formats like `jpeg` and `png`. However, when there is *a lot* of data points in a figure, and hence a huge amount of vectors is required to accurately represent the figure, vector graphics tend to become very large and very inefficient to generate and view. For such figures, raster graphics are usually preferrable – however, one would still like to achieve things like consistent font size and type between the figure and the surrounding document. One way to achieve this is by

[1]For importing data in pgfplots, these are best exported to a `csv` file, e.g. using `csvwrite` in `Matlab` or `numpy.savetxt` or `pandas.DataFrame.to_csv` in `Python`.



(a) (b)

Figure 2.3: The same data points, (a) plotted using `Matlab` and exported to an `.eps` figure file, and (b) exported to a `.csv` file from `Matlab` and plotted using `pgfplots`. Note how the font type and size of all labels in figure (b) agree nicely with those in the rest of the document.

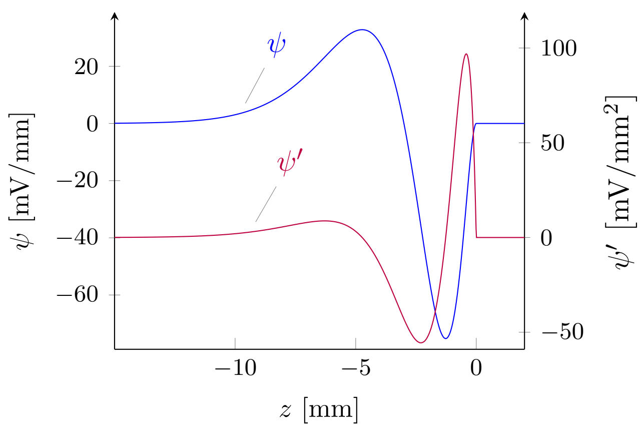


Figure 2.4: Plots of the intracellular action potential (IAP) model function proposed by Rosenfalck [12], generated using the `pgfplots` package. Figure reproduced from Petersen [8].

using the gnuplot software package. gnuplot essentially provides a simple scripting language that can be used interactively or script-based to generate, view, and export figures from scratch or using existing data. Figure 2.5 shows a rather sophisticated figure containing EMG signals that has been generated using gnuplot. For more information, refer to the accompanying script file scripts/emg-raw-plots.gp that generated this figure, and to the gnuplot homepage and various tutorials available on the web.

2.5 Referencing

Every statement that you make in your thesis should either be supported by evidence (theoretical or empirical) provided in your thesis, or by citation of an appropriate literature reference. In the present chapter (materials and methods) in particular, you should provide a lot of references, e.g., to scientific articles [2], books [9], book chapters [11], PhD theses [3], or software projects [10] that you used during the creation of your thesis. You must provide a reference whenever you make a statement that is not a result of your own (experimental or theoretical) work, but rather taken from a literary source. When selecting sources to cite, preference should be given to high-quality sources published in respected scientific journals (or books), wherever possible. Moreover, one should always try to cite the original source where a subject was discussed first or an algorithm was proposed first, instead of secondary sources. For example, regarding the infamous Kalman filter, one should always cite the original paper published by Kalman [6] instead of the many papers published later on (variations of) the same subject.

Sometimes, you should explicitly name authors, such as Farina and Rainoldi [2], who wrote a seminal article on the mathematical modelling of electromyography (EMG) measurements (see the Latex source document to see how this textual reference is created automatically)[2]. In some cases – especially when describing the fundamentals of your research – you will have entire paragraphs that are based on a single, significant reference on the subject you describe. In this case, it is neither necessary nor useful to cite said reference after each phrase. One acceptable solution is to have a phrase in the beginning of the paragraph or section that acknowledges

[2]Note how the EMG acronym is clickable and leads to the definition of this acronym in the glossary. This is one of the many features of the glossaries package.

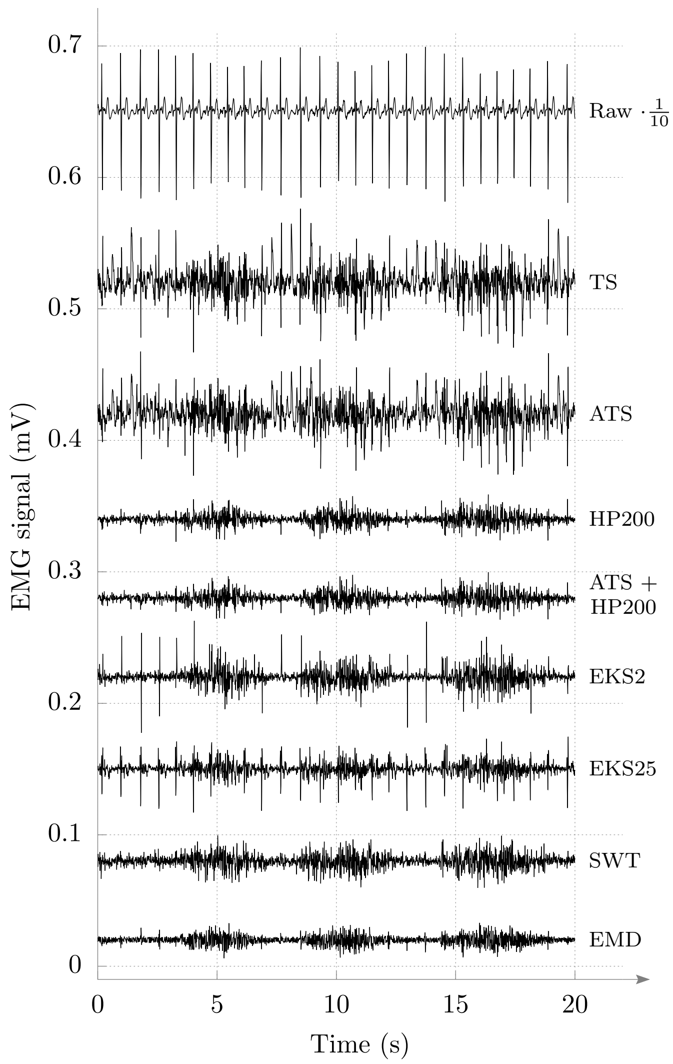


Figure 2.5: Exemplary subset of the measurements taken from subject 7, raw (plotted with a scaling factor of $\frac{1}{10}$) and processed by the various ECG removal algorithms described in the text. All signals are zero-mean, with offsets added for illustrational purposes only. Due to the huge amount of data points, representing this figure in a vector graphics format (using, e.g., `pgfplots`) would be very inefficient. Here, the figure has been produced using the `gnuplot` software in a raster graphics format, while still achieving nice and consistent visual formatting.

the source and states clearly that the entire following paragraph is based on this source.

Figures may be reproduced from sources, in which case the source must of course be acknowledged accordingly – refer to fig. 2.4 for an example of how this can be done. It is also considered good scientific practice to provide citations for the software packages that you have used, considering that these are often the result of many years of scientific work [10], [16]. Many software packages provide information on how to cite them on their project web pages.

Finally, a bibliography can be created automatically, based on the order in which you have cited sources (see the appendix for an example). This can be achieved using various tools and packages, which all require a separate file (often called `references.bib`) containing bibliographic information on the cited sources. In this template, the `biblatex` package in combination with the `biber` software is used. Usage is simple: After compiling your document once with `pdflatex`, a file `main.bcf` is generated, which essentially contains a list of the cited sources. Next, running `biber main` on the command line combines this generated list with the `.bib` file and generates a new file `main.bcf` that contains the required bibliographic information, using the correct order and citation style. Afterwards, run `pdflatex` again, to incorporate this bibliographic information in the generated document. This process has to be repeated every time a new source is cited in the main document, and each time bibliographic information in the `bib` file is added or changed. Note that there are various software packages available that enable easy handling of bibliographic data and automatic generation of a corresponding `bib` file, e.g. the open source package `JabRef`.

2.5.1 This is a subsection

See the source code for how this is achieved.

This is a subsubsection

Note that the different appearance of chapters, sections, subsection and subsubsections can be customized if desired.

2.6 Equations

If you are going to write more than a few equations in your thesis, it is highly recommendable to use macros to define each variable that occurs. This way, if you decide to rename the velocity from v to v_0 (see the source code of this section for how to include mathematical symbols like v in normal text) everywhere in your thesis later on during the process of writing, you will only have to change the definition of the macro used for this variable. For example, in

$$v = \frac{\mathrm{d}x}{\mathrm{d}t} \tag{2.1}$$

and

$$\begin{aligned} a &= \frac{\mathrm{d}v}{\mathrm{d}t} \\ &= \frac{f}{m}, \end{aligned} \tag{2.2}$$

each of the variables is defined using a macro. Always keep in mind that formulae should be treated as a normal part of a sentence and thus can and have to contain punctuation marks! *Never* should any sentence start with a mathematical symbol. Finally, here comes an example of how to reference equations (2.1) to (2.2). These references will update automatically if you decide to move these equations to a different part of the thesis, or if another equation is inserted between these two, hence changing equation numbering.

2.7 Pseudo code

Algorithm 1 is an example of how pseudo code can be represented in Latex. Of course there are, as with anything in Latex, many options available to achieve this goal. Here, the `algpseudocode` package is employed.

Algorithm 1 Euclid's algorithm

```
procedure Euclid(a, b)                      ▷ The g.c.d. of a and b
    r ← a mod b
    while r ≠ 0 do                          ▷ We have the answer if r is 0
        a ← b
        b ← r
        r ← a mod b
    end while
    return b                                ▷ The gcd is b
end procedure
```

3 Concept

This chapter describes the novelty that you have created during the making of your thesis. This novelty might be an experimental apparatus, a novel algorithm, or a thorough analysis of something. You may here reuse the background information presented in the previous chapter.

There is a wealth of guides available on thesis writing; one that I'd like to point out here because of its conciseness and direct availability is *Clarity in technical reporting* [7], an old yet still valid guide on the subject, originally created for internal use at NASA.

3.1 Thesis writing strategies

3.1.1 Have a plan before you start

Before starting to write any paragraph of full text in your thesis, you should know the contents of every chapter of your thesis to a high degree of detail. Otherwise, you will find yourself rewriting and maybe even dumping large parts of what you wrote earlier. Create a detailed outline of the story you want to tell in your thesis, and make it clear to yourself how each chapter contributes to this central story (also see below for general tips on structuring your thesis well). It also helps to collect and create explanatory figures early on that can help you identify the concepts you need to explain in your thesis.

Once you have a very detailed draft of your thesis *withoug having written any full sentences*, and you have discussed this draft in detail with your supervisor, doing the actual writing will be much, much easier than if you started right away. It will also reduce the chance of your supervisor telling you to rewrite large parts of your thesis. For a more detailed discussion of this writing strategy, also refer to Carlis [1].

3.1.2 Reproducible research

Reproducible research is a term that has gained a lot of attention recently. It describes research that is easily reproducible by yourself, as well as others. Note that many common research practices are not easily reproducible at all, e.g.

- Saving figures you somehow created interactively in `Matlab` – instead of saving a script file that reliably reproduces this figure.

- Saving statistics calculated from measured data, or modified data (e.g. smoothed signals, or with outliers removed) – instead of saving a script file that performs the desired operations.

- Changing a results table in your report according to new results obtained during interactive analysis – instead of saving a script file that automatically generates the data in the table, using the most current version of your software.

Keeping your work reproducible is not only useful for others, but also (and especially) for yourself: It makes sure that all parts of your work (data, analysis, report, etc.) are always in sync. Furthermore, you can, e.g., easily update all figures in your thesis a week before the deadline – which may be a nightmare if you created all of these figures manually and cannote reproduce them easily. A good introduction to the steps required to keep your work reproducible can be found at `http://kbroman.org/steps2rr/`. I do recommend checking it out right now!

3.2 Structure

The general structure of your text depends on the particular subject and your personal preferences. A reader following your text should at each point in your thesis feel like you are telling him a consistent story. In particular, ask yourself the following questions:

- What is your main contribution to the scientific community?

- Which concepts and background knowledge do you need to provide for the reader to be able to understand your contribution and its relevance?

- Which would be a sensible and natural ordering of the concepts that you need to explain?

Always keep in mind that the structure of your text should feel natural to a reader, not leaving him or her wondering what on earth this new topic has to do with anything he's read so far.

Start your scientific text with an introduction, that

- introduces the subject,
- specifies the topic,
- reflects on the problem that you are going to consider,
- defines the purpose of the work,
- explains the line of reasoning
- sketches the structure of the work.

Keep in mind that there are some readers that only read the introduction and conclusion of your work and base their decision on whether the work bears any interest for them only on these parts. To make that decision, they need to get all relevant information from those two chapters. Hint: Look at other work with a focus on that question.

For the main part of your work, there are no general rules. Among others, the order of your chapters should depend on whether your work has a rather theoretic, methodical or experimental focus. The length of each chapter does not necessarily need to be proportional to the amount of time you have spent on solving the respective problems. Sometimes it takes one week to debug a piece of code, which nevertheless should not be explained excessively.

Your work concludes with a summary of your results. Have a look at your introduction: how you have specified the problem there? Do your results solve the problem? Do not present any further results here that have been not presented in the main part. As such, always clearly separate the presentation and the discussion of results.

Finally, you end with an outlook that points out open questions. What should be further analyzed and what are possible follow-up projects? Do not be afraid to point out questions that came to your mind during your research, but you did not have time to properly answer. A good thesis may raise more questions than it clarifies.

3.3 Language

For a very good and readable (and cheap) guide on improving your writing skills, refer to Zinsser [17]: *On Writing Well.* Every minute you spend reading this book will be well worth its time.

3.3.1 Correctness

Scientific writing serves a single purpose: to transport information as efficiently as possible. To achieve this, your writing should be as accurate as possible. Do not use colloquial language. Respect the rules of grammar, spelling and punctuation; phrases or words used in the wrong context can lead to misunderstandings or may be hard to understand. It should be clear that the results of your work, e.g., experiments, must be documented precisely and correctly, even though they might have had unexpected outcomes. Otherwise you do not only cause harm to you but any further research.

3.3.2 Comprehensibility

Correctness does not imply comprehensibility. Look at your text from the readers point of view: Consider his or her position, previous knowledge and attitude. Formulate as precisely as possible but not more complicated than necessary. Therefore,:

- choose words, that are known;
- if you have to use words that are probably unknown, use them such that their meaning can be deduced from the context, or explain or define them;
- do not construct deeply nested sentences.

To achieve clarity and conciseness in your writing, you may find the following method very useful (I use this whenever I write *any* text). For any paragraph you have written, ask yourself the following questions:

- Which information or logical argument do I want to convey to my reader using this paragraph? Which are the different steps of this argument, and in which order to they make sense?
- Does the paragraph have a clear focus on a single subject, or do you write about a number of different things? (If the latter is the case, separate the contents into multiple paragraphs.)
- Are there logical „gaps“ in the paragraph, i.e., are there steps missing to be able to understand the described argument? (If yes, insert them.)
- Do I currently convey this information in the most effective way? Are there phrases that could be shortened or removed without hurting the paragraph’s main message? (If yes, do so.)
- Does the paragraph „flow“ naturally from one phrase to the other, following a logical progression? (If not, think again about the steps of the logical argument you are trying to give, and try to emphasize the logical connections between consecutive phrases.)

Many example applications of this method can be found in the book of Zinsser recommended above. While it takes a while to get used to this way of thinking about writing, I find this method tremendously useful for communicating complex subjects effectively. Note that the same questions can also be asked both on a smaller (single sentence) as well as larger scale (whole section or chapter). Doing so can really help to strip down your text to the absolutely necessary minimum required to convey the main results of your work, which will greatly improve the clarity and conciseness of your thesis.

3.4 Preparing a scientific presentation

Generally, the same guiding principles that apply to writing a scientific report also apply to the preparation of a scientific presentation: A meaningful structure, clarity, conciseness and correct presentation of contents, including correct referencing of previous works of other scientists. However, in a presentation, the focus should be strongly shifted away from using a lot of text, towards the display of informative figures and diagrams that visually support your oral presentation of concepts and results.

A recent trend in scientific presentation styles has been to move away from the classical „Introduction - Methods - Results - Conclusion“ structure of presentations, and more towards what is called an „Assertion - Evidence“ structure. In this style, slides are usually composed of a title line that displays a meaningful message you would like to convey to your audience („Respiratory Mechanics can be modeled as Electrical Circuits“), instead of a classical caption („Modelling of Respiratory Mechanics“), and supporting visual evidence (i.e., a mechanical and an electrical model of respiratory mechanics), instead of bullet point lists. All further information required to understand the slide is presented orally by the speaker. Scientific studies have shown this style of slides to be more effective in conveying information to the audience than classically designed slides. For more information on this presentation style (and pointers towards the mentioned studies), refer to `http://www.assertion-evidence.com/`.

The following is just a further quick link list compiled to help in creating good scientific presentations.

- `http://www.the-scientist.com/?articles.view/articleNo/28818/title/Pimp-your-PowerPoint/`
- `http://www.northwestern.edu/climb/resources/oral-communication-skills/creating-a-presentation.html`
- `http://www.nextscientist.com/improve-presentation-skills-of-phd-students/`

4 Results

This chapter summarizes the results that you have obtained during experimental validation of the concept presented in the previous chapter. Note that a clear separation between experimental results and their discussion is to be made: Here, only numerical results, performance plots, etc. are to be shown and described. Any interpretation of these results – What do they imply? Which conclusions do you draw? – is to be postponed to the following chapter. This is useful since in the future, some of your interpretations may be clearly known to be false, due to newly acquired knowledge. In this case, however, the data you have collected during your studies may still be very valuable and hence should be easily accessible on its own. Moreover, this provides a very important distinction between objective findings, i.e. experimental measurements that you have obtained or other data, and the inferences that you draw from these data, which may or may not be true.

5 Discussion

Finally, in this chapter, the results presented in the previous chapter are to be discussed and interpreted. Here you may discuss the prospects of your proposed method to solve the problem at hand, and propose alternative solutions.

6 Conclusion

In this final chapter, you provide a concise summary and conclusion of your work.

Acronyms

EMG	Electromyography
IAP	Intracellular Action Potential
SFAP	Single Fiber Action Potential

List of Mathematical Symbols

a	Acceleration
f	Force
m	Mass
v	Velocity
x	Location, spatial variable

List of Figures

Bibliography

[1] J. Carlis, "The key to writing a one-draft ph.d dissertation", 2009. [Online]. Available: `www-users.cs.umn.edu/~carlis/one-draft.pdf`.

[2] D. Farina and A. Rainoldi, "Compensation of the effect of sub-cutaneous tissue layers on surface EMG: A simulation study", *Medical Engineering & Physics*, 1999.

[3] C. Févotte, "Approche temps-fréquence pour la séparation aveugle de sources non-stationnaires", PhD thesis, École Centrale de Nantes et Université de Nantes, 2003.

[4] S. Few, *Show Me the Numbers: Designing Tables and Graphs to Enlighten.* Analytics Press, 2012.

[5] *Fixme – collaborative annotation tool for latex*, 2017. [Online]. Available: `http://mirrors.ctan.org/macros/latex/contrib/fixme/fixme.pdf`.

[6] R. E. Kalman, "A new approach to linear filtering and prediction problems", *Journal of Basic Engineering*, vol. 82, no. 1, p. 35, 1960. DOI: `10.1115/1.3662552`.

[7] S. Katzoff, "Clarity in technical reporting", Tech. Rep., 1964. [Online]. Available: `http://courses.media.mit.edu/2011spring/mas111/NASA-64-sp7010.pdf`.

[8] E. Petersen, "Integrative mathematical modeling of surface electromyographic modeling", Master's thesis, University of Hamburg, 2015.

[9] R. Plonsey and R. C. Barr, *Bioelectricity: A Quantitative Approach*, Third Edition. Springer, 2007.

[10] R Core Team, *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, Vienna, Austria, 2014. [Online]. Available: `http://www.R-project.org/`.

[11] J. Rodriguez-Falces, J. Navallas, and A. Malanda, “EMG modeling”, in *Computational Intelligence in Electromyography Analysis - A Perspective on Current Applications and Future Challenges*, G. R. Naik, Ed., InTech, Oct. 2012.

[12] P. Rosenfalck, “Intra and extracellular fields of active nerve and muscle fibers. a physico-mathematical analysis of different models”, *Acta Physiologica Scandinavica*, 1969.

[13] N. P. Rougier, M. Droettboom, and P. E. Bourne, “Ten simple rules for better figures”, *PLoS Computational Biology*, vol. 10, no. 9, e1003833, Sep. 2014. DOI: `10.1371/journal.pcbi.1003833`.

[14] E. Tufte, *Envisioning Information*. Graphics Press, 1990.

[15] C. Ware, *Visual Thinking for Design*. Morgan Kaufmann, 2008.

[16] H. Wickham, *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2009, ISBN: 978-0-387-98140-6. [Online]. Available: `http://ggplot2.org`.

[17] W. Zinsser, *On Writing Well*, 7th. Harper Perennial, 1976.

List of Corrections