



TECHNICAL UNIVERSITY OF MUNICH

DEPARTMENT OF INFORMATICS

Bachelor's Thesis in Informatics

Realistic Optimization-based Driving Using a Constrained Double-Integrator Model

Andreas Belavic



TECHNICAL UNIVERSITY OF MUNICH

DEPARTMENT OF INFORMATICS

Bachelor's Thesis in Informatics

Realistic Optimization-based Driving Using a Constrained Double-Integrator Model

Realistisches, optimierungsbasiertes Fahren mit einem beschränkten Doppel-Integrator-Modell

Author:	Andreas Belavic
Supervisor:	Prof. Dr.-Ing. Matthias Althoff
Advisor:	M.Sc. Tobias Mascetta, M.Sc. Lukas Schäfer
Submission Date:	01.01.2019

I confirm that this bachelor's thesis is my own work and I have documented all sources and material used.

Ich versichere, dass ich diese Bachelor's Thesis selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

Munich, 01.01.2019

Andreas Belavic

Acknowledgments

A thesis may include an acknowledgments section where the author may want to thank certain people, groups, institutions, etc. who provided support.

Abstract

This is the abstract. It is a short summary of your work, consisting of roughly one to three paragraphs. It should give the main ideas of your paper, i.e., the posed problem, a motivation for solving it, your solution method, and your results results results. Keep it understandable for a general audience. Do not include references.

Contents

Acknowledgments	iii
Abstract	iv
1 Introduction	1
1.1 Submission	1
1.2 Overview	2
2 Format	3
2.1 Introduction	3
2.2 Approach	4
2.2.1 Implementation/Method	4
2.3 Numerical Experiments	4
2.4 Conclusion	5
3 Style	6
3.1 General Information	6
3.2 Punctuation	6
3.2.1 Commas	6
3.2.2 Semi-Colon	7
3.3 Parts of Speech - Common Errors	8
3.3.1 Adverbs	8
3.3.2 Verbs	9
3.3.3 Person	10
3.4 Regional Differences	10
4 Latex	11
4.1 Using this Template	11
4.2 Figures	11
4.3 Math	12
4.4 Tables	12
4.5 Algorithms	13
4.6 Citation	13

Contents

5 Literature Review	14
List of Figures	15
List of Tables	16

1 Introduction

The *student starter kit* serves as guidance for students writing a thesis. This LaTeX document contains some advice on writing and can be used as template for your thesis. In addition, this starter kit contains a template for presentations, standardized grade sheets used to transparently mark your work, and a template for submitting your files (see Sec. 1.1). As a short summary, the kit also contains two text files: a checklist for your thesis and one for your presentation; please adhere to them!

The student starter kit is only to be used within the Professorship Cyber-Physical Systems. The kit has been created by Markus Koschi¹. This latex template is based on the template by Florian Walch and contributors². The advice on format and style are from Carmella Schürmann. The advice on the literature review are from Matthias Althoff.

Further reading is referenced where applicable, and general information provided by TUM is available in German³ and English⁴. The advice given by Elmar Juergens, a former PhD student at TUM in software engineering, might be also useful for you: <https://thesisguide.org/>.

1.1 Submission

For internal archiving, you are required to submit all source files of your thesis, presentation, and software in the given format (see the enclosed submission folder): The parent folder should be named `firstNameLastName` and contain the following sub-folders:

- `papers_FirstNameLastName` (folder containing studied publications)
- `presentation_FirstNameLastName` (folder for the pdf and source files of the presentation)
- `report_FirstNameLastName` (folder for the pdf and source files of the report)

¹markus.koschi@tum.de

²<https://github.com/fwalch/tum-thesis-latex>

³<https://www.tum.de/studium/studienabschluss/abschlussarbeit/tipps-und-tricks/>

⁴<https://www.tum.de/nc/en/studies/completing-your-studies/theses/tips-and-tricks/>

- software_FirstNameLastName (folder for the source files of the created software)

Please submit this folder at latest on the day of your presentation either via mail or personally by flash drive. The pdf of your thesis is required on the official day of submission. Note that you should aim at reproducible results, i.e., that your advisor can run your code and reproduce your results. Thus, providing a read-me and saving the simulation parameters in a script might be a good idea.

1.2 Overview

This document explains how to structure a thesis in Ch. 2 and how to write well in Ch. 3. Ch. 4 provides advice on how to use Latex, and Ch. 5 describes best practices to conduct a literature review.

2 Format

This chapter introduces the basic layout of your thesis. In general, you want to make sure that each chapter flows smoothly and is linked to the one before. One way to do this is to briefly (in half a sentence or less) summarize what was discussed in the previous chapter, though you do not want to become too repetitive. If this cannot be done well, then leave it out. More critical is a short overview of what your reader can expect in the current chapter. Consider the possibility that your reader may not only read your work from start to finish in one go, but may also re-read certain chapters individually or out-of-order.

- *In the previous chapter, xyz was discussed. Now we turn to.../Now we will look at.../In this chapter...*

What follows is a general outline, although there is no one “correct” way to format your thesis. The sections given in this reference guide (Introduction, Approach, Evaluation, etc.) each refer to a different chapter of your thesis. The subsections presented here correlate therefore to chapter sections. Depending on what makes sense for your topic and how you formulate your argument, certain parts can be merged or split. As mentioned above, make sure to tell a complete story from beginning to end, i.e., structure each chapter as a mini-thesis with an introduction, main part, and conclusion (though not necessarily divided that way into sections). For more information, see e.g. this website¹.

2.1 Introduction

The first chapter of your text. Clearly state your problem, what your work attempts to do, as well as why it is important. This should be understandable for a general audience, as you do not know who will be reading your thesis.

Present the related work. Explain what has already been accomplished/solved – and then explain what your approach offers that previous ones do not. What are their disadvantages, and how does your approach improve upon their drawbacks? The literature review section should summarize and evaluate related works, thus proving

¹<https://student.unsw.edu.au/thesis-structure>

to your reader that your presented problem exists. To find related work, see the advise presented in Ch. 5.

Once you have presented the previous approaches, restate your approach, its novelty, and the methods you use. Present the challenges of your approach as well – you are not going to save all the world’s problems with your thesis, but you can try to make it easier to do so. Do not make sweeping claims and over-generalizations, e.g., *This topic is the most important of this generation and many to come.*

End the introduction with an outline of your paper. Introduce each chapter/section and give a brief summary of what they contain. *This paper/thesis/work is structured as follows: In Chapter 2...is presented in Chapter 3....Chapter 4 discusses...*

Often, it is useful to write the introduction after you have completed the rest of the paper: that way, you can correctly summarize what you set out to accomplish in the first place and how you did so. However, you should review related work as soon as you start your thesis to not reinvent the wheel.

2.2 Approach

This can be considered the main part of your work. Here, you want to go into detail about your approach. Explain the concept and methods used, as well as any shortcomings it may have.

2.2.1 Implementation/Method

Depending on your approach, a description of your implementation may merit its own section or subsection. If the approach you are presenting is more general and/or can be applied to different areas, you may only have the time and resources for one example implementation.

This section should therefore describe your implementation process thoroughly and precisely, giving all necessary parameters, so that your reader can reproduce your results. If necessary, your source code can go in the appendix.

2.3 Numerical Experiments

Here, you should give the results of your implementation and critically discuss them in detail. In order to prove the applicability/relevancy of your thesis, you typically also need to compare your approach to others. You should clearly define the setup and parameters of your experiment, then give your results and discuss them in detail. If your results are unexpected, figure out why and explain that to the reader.

2.4 Conclusion

Here is the place to summarize what you have written: your goal, your approach, your results. Discuss what can be drawn from what you have achieved. End the thesis by describing your future work, i.e., what from your approach can be expanded and improved upon in further projects.

3 Style

This chapter provides some information and tips about writing style. As TUM student, you can benefit from free consulting of the English Writing Center¹, which are highly recommended so that your scientific advisor can focus her/his support on the content of your work (rather than having to polish language first).

3.1 General Information

Keep it simple. Your reader should be able to reach the end of a sentence and remember how it began. If your sentences are long and complicated, see if you can break them up.

3.2 Punctuation

English punctuation can be tricky. Many rules are steadfast – there are no exceptions. Others, however, vary based on which style manual one uses. In general, there are two preferred styles for technical fields like computer science: that of the **Institute of Electrical Engineers (IEEE)** and that of the **Association for Computing Machinery (ACM)**. You can find more information about their citation formatting on their respective websites or in Section 4.6. The **IEEE** style falls under the broader Chicago Manual of Style, so the answers to punctuation and syntax questions can be found there. Note that these all use American formatting standards.

3.2.1 Commas

Some typical rules regarding commas include:

1. You **must** put a comma between two **independent clauses** (i.e., complete sentences) which are joined by a conjunction, *for*, *and*, *nor*, *but*, *or*, *yet*, *so*.
 - The exception to this rule is very short sentences – *I computed A and my partner computed B*. It is unlikely that you will encounter such sentences in your writing, so this exception can be ignored 95% of the time.

¹<https://www.sprachenzentrum.tum.de/sprachen/englisch/english-writing-center/>

2. To avoid confusion, use the serial/Oxford comma, i.e., the final comma in a list with three or more items.
 - For example: We present our method, our results, and a discussion.
3. Never put a comma between *to be* and *that*. Likewise, do not put a comma before any form of *to be*.
 - Incorrect: The biggest confusion, is that there is not enough data.
 - Incorrect: The biggest confusion is, that there is not enough data.
 - Correct: The biggest confusion is that there is not enough data.

*There may be instances when you *should* put a comma before a form of *to be*, but this will serve a different grammatical function, e.g., to set off an appositive. An appositive is a noun or pronoun which is set next to another noun or pronoun in order to explain it. For example: "The conclusion, the part of the paper which summarizes everything, is the trickiest part to write."
4. Put a comma before "which" if the text that comes before it can be understood without the text that comes after it. Otherwise, do not.
 - For example: We use the xyz control method, which was presented in Chapter 2.
 - Here, the fact that this method was presented in Chapter 2 is additional information. You can understand the sentence without it.
 - We use the control method which was presented in Chapter 2.
 - Here, you need to define which control method you are referring to; thus, that information is vital to properly understanding the sentence.
5. Use a comma when beginning sentences with introductory words such as "well", "now", or "yes".
6. According to the IEEE guidelines, you should put a comma after "i.e." and "e.g." This is widely accepted by most American style guides. It is less common in British English.
7. For more comma rules, see the enclosed document `more_comma_rules.pdf`.

3.2.2 Semi-Colon

The semi-colon has two main functions. It can be used to join two complete sentences whose meanings are closely related: "The results are inconclusive; this is due to the fact that the data does not represent all possibilities."

Additionally, the semi-colon can be used as a sort of “super-comma” when lists have internal punctuation: “This section has an introduction, which gives an overview; a main part, where the methods are discussed; and a conclusion.”

3.3 Parts of Speech - Common Errors

In this section, common errors/confusing rules will be discussed.

3.3.1 Adverbs

What is an adverb, again? An adverb modifies a verb, an adjective, another adverb, or even a phrase. It typically, although not always, ends in “-ly.”

Placement

Modifying Adjectives

When adverbs modify an adjective, they should be placed directly before the word they are modifying. For example, “Table 2 shows *similarly promising* results.”

Modifying Sentences

When modifying sentences, adverbs can be placed in four different positions:

- at the beginning: ***Usually**, such experiments are doomed to fail from the start.*
- at the end: *We then repeat these steps **frequently**.*
Adverbs appearing at the end of the sentence include:
 - adverbs of frequency - *usually, normally, often*, etc.
 - adverbs of manner, i.e., how - *slowly, badly*
 - adverbs of time - *today, this year*
- after *to be* and auxiliary verbs, e.g., *can, may, will*, etc.: *The problem will **definitely** be fixed.*
- before all other verbs: *We **quickly** solve the equation.*

3.3.2 Verbs

Tense

The question of tense can be rather complicated. You may have learned that one should always use the present tense, even when referring to past literature. Different styles have different rules, though often these “rules” are simply suggestions. The main body of your text should be written in the present tense. However, your literature review may use different tenses, depending on what you wish to express.

The generally accepted guidelines on using tenses in the literature review are as follows:

- Simple Past: Use the simple past tense (*found, reported, introduced*) when you are describing the results/methods of an already published paper. This tense is used when the citation/source is the subject of the sentence: “Smith [1] found that this method was not very successful.”
- Present: Use the present tense in the literature review if you are offering your own opinion of the literature. For example, “Smith [1] argues that this approach does not make sense; however, it is clear that the methodology had some flaws.”
- Present Perfect: This is typically used to indicate that the research you are citing is still ongoing/relevant, or that your source is quite recent. It can also be used to generalize about past literature. - “Similar findings have been discussed more recently [7], [8], [10].”

Voice

As with tense, selecting which voice (active or passive) has advocates in both corners. Some writers will tell you to avoid the passive voice at all costs. Others will say that scientific writing uses the passive voice much more often than other disciplines. Overusing the passive voice can get dry though, so it is generally good to vary your sentences.

- Active Voice:
 - “We show the results in Table 6.”
 - “We then compute the following equation: ...”
- Passive Voice:
 - “The results are shown in Table 6.”
 - “The following equation is then computed: ...”

3.3.3 Person

Do not use the first person singular (“I”) in an academic paper. Instead, use “we” or use the passive voice.

3.4 Regional Differences

Unless otherwise stated, you are free to choose between American and British English. Once you have done so, stay consistent and stick to the chosen style. Set your spell-check accordingly. Typical spelling differences between American and British English include:

- “o” vs. “ou”: *behavior* (Am.) vs. *behaviour* (Br.)
- “er” vs. “re”: *center* (Am.) vs. *centre* (Br.)
- “z” vs. “s”: *analyzed* (Am.) vs. *analysed* (Br.)
- double consonants: *modeled, modeling* (Am.) vs. *modelled, modelling* (Br.)

Reliable dictionaries are Merriam Webster² (American) and the Oxford English Dictionary³ (British).

²<https://www.merriam-webster.com>

³<https://en.oxforddictionaries.com>

4 Latex

After learning about the content of a thesis, we will discuss how to present content using Latex. If you are familiar with Latex, you can skip this chapter; others will serve this as a time-saving starting point.

4.1 Using this Template

This template is for your reference only. You do not have to adhere to this format and can change anything you like.

To use this template, configure some settings in the settings folder. Then, add your content in the content folder (you can use multiple `.tex` files, but it might be easier to only use one) and your references to `literature.bib` (see also Section 4.6).

You can compile a pdf from `main.tex` either using *pdflatex* or *latex + dvips + ps2pdf* (often configured as quick build). In an editor, e.g. Texmaker, you can set `main.tex` as master document, and thus compile from any file.

4.2 Figures

Figures are a very important tool to present content. A (good) visualization often helps the reader to understand your concepts. But do not forget to also explain in words what can be seen in the figure.

If you use *pdflatex*, you can include pdf figures, and if you use *latex*, you can include eps figures (please always use vector graphics). To draw figures or edit plots you generate with MATLAB or Python, you can use Inkscape¹.

Fig. 4.1 demonstrates how we include a figure. If you use *latex + dvips + ps2pdf*, the eps is included and *psfrag* replaces the letters *a*, *b*, and *c* with latex code. If you use *pdflatex*, you can save a pdf figure from Inkscape with the option to omit text in the pdf and create an additional `pdf_tex` file, whose text will then be typeset with Latex (not shown here).

¹<https://inkscape.org/en/>

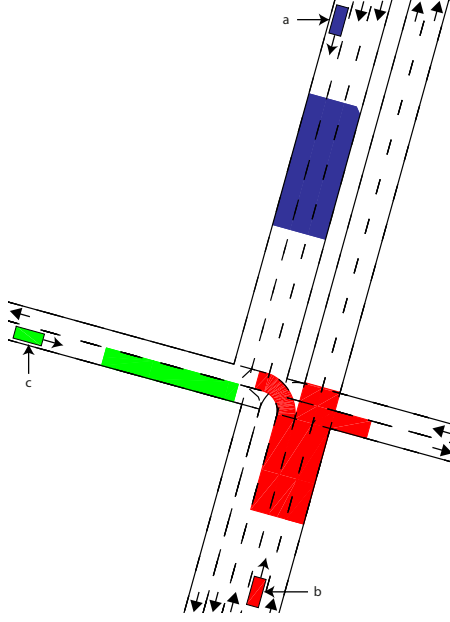


Figure 4.1: Occupancies of Obstacles 1, 2, and 3 in Scenario I. The plot shows the initial configuration at t_0 and the predicted occupancies $\mathcal{O}(t)$ for $t \in [t_3, t_4]$.

4.3 Math

As an example, let us introduce $\mathcal{X} \subset \mathbb{R}^n$ as the set of feasible states x and $\mathcal{U} \subset \mathbb{R}^m$ as the set of admissible control inputs u of a system f , which is governed by the differential equation

$$\dot{x}(t) = f(x(t), u(t)). \quad (4.1)$$

We assume that the initial time is $t_0 = 0$ and adhere to the notation $u([0, t_h])$ to describe a trajectory $u(t) \in \mathcal{U}$ for $t \in [0, t_h]$, $0 < t_h$. Furthermore, $\chi(t_h, x(0), u([0, t_h])) \in \mathcal{X}$ denotes the solution of (4.1) at time t_h subject to $x(0) = x_0$ and $u([0, t_h])$.

Note that you should not reference an equation before introducing it. After giving the equation, you can refer to it, e.g., our system (4.1) is awesome.

4.4 Tables

Tab. 4.1 is an example of a nice table.

Table 4.1: Obstacle Parameters.

Parameter	Variable	Parameter	Variable
max. acceleration	a_{\max} in m/s^2	Boolean for C_{back}	b_{back}
max. velocity	v_{\max} in m/s	Boolean for C_{lane}	b_{lane}
switching velocity	v_S in m/s	length	l in m
speeding factor	f_S	width	w in m

4.5 Algorithms

Often, it will be helpful to describe your software by presenting pseudo code and explaining it (at least the significant lines).

Alg. 1 shows the overall algorithm of *SPOT*, which predicts the occupancy for a traffic participant up to the prediction horizon and can run in parallel for each traffic participant. First, the constraint management configures the parameters according to the set of last-measured states of the obstacle (line 1) as detailed in the next paragraph...

Algorithm 1 Occupancy Prediction for an Obstacle

Require: map

- 1: parameters \leftarrow MANAGECONSTRAINTS
 - 2: reachableLanes \leftarrow REACH(map, parameters)
 - 3: **for all** VALIDABSTRACTIONS(parameters) **do**
 - 4: $\mathcal{O}_j \leftarrow$ OCCUPANCY_j(reachableLanes, parameters)
 - 5: **end for**
 - 6: **return** $\mathcal{O} \leftarrow$ INTERSECT(\mathcal{O}_j)
-

4.6 Citation

For citations, we use a consecutively numbered citation style: bla [latex] bla [Koschi2017a, Althoff2017a] bla [Paden2016]. To generate a bibliography file (the .bib file located in the folder content), you can use JabRef². Then, you run *bibtex* and afterwards latex/pdflatex twice (as described in Sec. 4.1).

²<http://www.jabref.org/>

5 Literature Review

To find relevant works, you should follow this guideline:

- Important catalogues: IEEE XPlore, Google scholar, Scopus, Web of Science, Research Gate
- Start searching catalogues: Write down used keywords and used catalogues to avoid repeating a search; try some searches with the keyword “survey” or “overview” to find survey papers
- Group results: Find categories in which to put your found papers
- Rank papers in each category: consider reputation of journal/conference, reputation of authors, and citations
- In best ranked papers: Look into their literature lists; what other papers are relevant (do this iteratively)? Newer papers are more useful since they cover the literature up to a later point in time. Finding a great and recent survey paper is the jackpot
- Once the search has converged: identify the most important groups
- Visit websites of important groups to identify missed papers

List of Figures

- 4.1 Occupancies of Obstacles 1, 2, and 3 in Scenario I. The plot shows the initial configuration at t_0 and the predicted occupancies $\mathcal{O}(t)$ for $t \in [t_3, t_4]$. 12

List of Tables

4.1	Obstacle Parameters.	13
-----	------------------------------	----