

# Model identification and learning control in autonomous driving

Stijn Staring

Thesis voorgedragen tot het behalen  
van de graad van Master of Science  
in de ingenieurswetenschappen:  
elektrotechniek, optie Elektronica en  
geïntegreerde schakelingen

**Promotor:**

Prof. dr. ir. Jan Swevers

**Assessoren:**

Prof. dr. ir. Bert Pluymers  
Prof. dr. ir. Herman Bruyninckx

**Begeleider:**

dr. ir. Son Tong

© Copyright KU Leuven

Without written permission of the thesis supervisor and the author it is forbidden to reproduce or adapt in any form or by any means any part of this publication. Requests for obtaining the right to reproduce or utilize parts of this publication should be addressed to ESAT, Kasteelpark Arenberg 10 postbus 2440, B-3001 Heverlee, +32-16-321130 or by email [info@esat.kuleuven.be](mailto:info@esat.kuleuven.be).

A written permission of the thesis supervisor is also required to use the methods, products, schematics and programmes described in this work for industrial or commercial use, and for submitting this publication in scientific contests.

Zonder voorafgaande schriftelijke toestemming van zowel de promotor als de auteur is overnemen, kopiëren, gebruiken of realiseren van deze uitgave of gedeelten ervan verboden. Voor aanvragen tot of informatie i.v.m. het overnemen en/of gebruik en/of realisatie van gedeelten uit deze publicatie, wend u tot ESAT, Kasteelpark Arenberg 10 postbus 2440, B-3001 Heverlee, +32-16-321130 of via e-mail [info@esat.kuleuven.be](mailto:info@esat.kuleuven.be).

Voorafgaande schriftelijke toestemming van de promotor is eveneens vereist voor het aanwenden van de in deze masterproef beschreven (originele) methoden, producten, schakelingen en programma's voor industrieel of commercieel nut en voor de inzending van deze publicatie ter deelname aan wetenschappelijke prijzen of wedstrijden.

# Preface

I would like to thank my family in the first place. During the history of my studies they always have been my biggest fans and I want to show my gratitude for the opportunities they have given me. I also want to thank my promoter Professor Swevers at the KU Leuven and Dr. Tong my mentor at Siemens for the professional discussions and tips they have given me in order to improve results.

*Stijn Staring*

# Contents

<b>Preface</b>	<b>i</b>
<b>Abstract</b>	<b>iii</b>
<b>Samenvatting</b>	<b>iv</b>
<b>List of Figures and Tables</b>	<b>v</b>
<b>List of Abbreviations and Symbols</b>	<b>vi</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 State of the art modelling of comfort</b>	<b>5</b>
2.1 What are the parameters that define comfort during driving? . . . .	5
2.2 Inverse reinforcement learning . . . . .	7
<b>3 The Learning Algorithm</b>	<b>11</b>
3.1 The First Topic of this Chapter . . . . .	13
<b>4 Background in optimal control problems</b>	<b>15</b>
<b>5 Path Planning MPC</b>	<b>21</b>
<b>6 Path tracking MPC</b>	<b>23</b>
6.1 The First Topic of this Chapter . . . . .	23
6.2 The Second Topic . . . . .	23
6.3 Conclusion . . . . .	23
<b>7 Validatie</b>	<b>25</b>
7.1 The First Topic of this Chapter . . . . .	25
7.2 The Second Topic . . . . .	25
7.3 Conclusion . . . . .	25
<b>8 Conclusion</b>	<b>27</b>
<b>A The First Appendix</b>	<b>31</b>
<b>B The Last Appendix</b>	<b>33</b>
<b>Bibliography</b>	<b>35</b>

# Abstract

The **abstract** environment contains a more extensive overview of the work. But it should be limited to one page.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

# Samenvatting

In dit **abstract** environment wordt een al dan niet uitgebreide Nederlandse samenvatting van het werk gegeven. Wanneer de tekst voor een Nederlandstalige master in het Engels wordt geschreven, wordt hier normaal een uitgebreide samenvatting verwacht, bijvoorbeeld een tiental bladzijden.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

# List of Figures and Tables

## List of Figures

1.1	Concept visualization of autonomous driving. (source: [6]) . . . . .	1
1.2	Example lane change as used as input in the inverse optimal control algorithm. . . . .	3
2.1	Overview of comfort parameters in autonomous vehicle with old parameters (blue) and new ones (red).(Source: [7]) . . . . .	6
4.1	Overview of different discretization methods. . . . .	16
4.2	Schematic view of the time shooting approaches (left: multiple shooting; right: single shooting). . . . .	17
4.3	Visualization of the optimal control problem solved in one iteration of the MPC (Source: [13]). . . . .	18
4.4	Visualization of the application of the first step of the calculated control signal during one iteration of the MPC (Source:[13]). . . . .	18

## List of Tables

# List of Abbreviations and Symbols

## Abbreviations

LoG      Laplacian-of-Gaussian

## Symbols

42      “The Answer to the Ultimate Question of Life, the Universe, and Everything”  
according to [?]  
 $c$       Speed of light



# Chapter 1

## Introduction

### 1.0.1 Importance of topic

"Society expects autonomous vehicles to be held to a higher standard than human drivers." [16] This quote is setting the tone of the technology in autonomous driving. In order to be accepted to the public, autonomous vehicles should perform as least as good as the conventional human driver on parameters as for example safety. Despite widespread research on self-driving vehicles the acceptance by the user stays only limited.[2] The purchase behaviour of customers can be directly linked with comfort. Also in order to gain more trust by the public it is clear that the challenge of making autonomous vehicles as comfortable as possible, should be tackled. This immediately leads to the questions what comfort during driving exactly is and how to measure it. A survey was conducted by researchers of the university of Warwick with as research question: Do passengers prefer autonomous vehicles be driven as full efficiency machines or in a way that emulates averaged human behaviour? [20] The result suggested that a blend of both is appealing the most confidence by user of a autonomous vehicle.

Driving comfort is a personal experience and also depend on the current emotional state of the driver. This means that more than one driving style for autonomous vehicle-driving should be identified for a certain vehicle. [22] The state of the driver can be communicated with the vehicle at the start of each ride and different driving styles can be obtained by changing the parameters in the path planning algorithm.



FIGURE 1.1: Concept visualization of autonomous driving. (source: [6])

### 1.0.2 Link with previous studies and problem formulation

In order to identify the specific comfort preferences of the driver that are quantized by these parameters, the vehicle should be able to learn them by demonstration. [11] Despite that each driver has its own preferences, they are based on a common notion of comfort where only different trade-offs are made. For example some drivers prefer more aggressive driving behaviour than others which will manifest itself in a different trade-offs of different comfort criteria than for example a defensive drivers. This will later in this thesis be translated into a comfort objective where different weights are used in order to quantify different comfort trade-offs made. This approach is in literature called inverse optimal control because it is learning the objective function of the comfort optimization. The lower the outputted value of this comfort objective, the higher the measurement of attained comfort will be in the later path planning algorithm.

In order to find comfort criteria which can be used to distinguish different drivers, research about the common notion of comfort is necessary. Passenger surveys in public road transport about carsickness [19] have identified lateral acceleration as the primarily responsible for motion sickness. It is explained that drive style is a main factor to influence the amount of sickness and it was found that sickness is higher when drivers drove with a higher average magnitude of fore-and-aft and lateral motion. These effect were found far more significant than the effect of vertical vibrations. There is also a consensus reached about the contribution of continuous trajectories to the prevention of motion sickness and the natural feel of paths.[7] This means that higher order kinematic variables like accelerations and jerks also should be considered when measuring comfort.

### 1.0.3 Thesis objective

The goal of this thesis is to build further on the research of learning by demonstration [11] and to refine this idea in a good working practical application. The thesis is more concretely focussed in the ability to explain driver data and the practical implementation and validation of a learning algorithm that is able to capture user specific driving preferences in weights of a comfort objective function. The learning process is to be done offline and is based on an inverse optimal control approach. A comfort objective will be derived from literature to describe the common notion of comfort and this will be fitted on individual driver data to produce driver specific parameters. In the next step this objective function will be used in a path planning MPC formulation which will be calculating online feasible and comfortable paths whereafter an tracking MPC algorithm will follow it.

To conduct the inverse learning control there is first look at data generated by simulations where it is assumed that the vehicle is driving on an straight road and high way speed when executing manoeuvres as lane changes and longitudinal accelerations.

---

An example lane change can be seen in Figure 1.2. Assumptions made during this manoeuvre To be able to make the generated data of high quality an MPC approach with a 15 degree of freedom vehicle model is used. Also in the learning algorithm itself a three degree of freedom non-linear bicycle model is used in order to adequately capture the different kinematic signals e.g. jerks and accelerations. Further there were comparisons made of different methods to learn from multiple datasets.

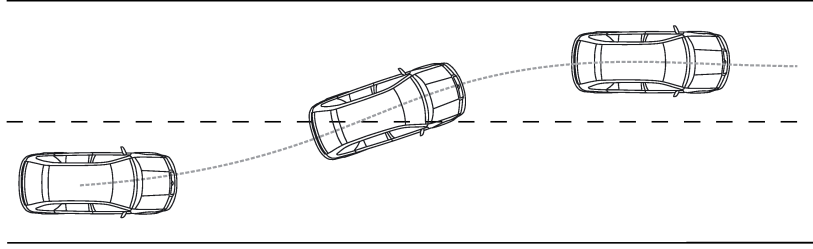


FIGURE 1.2: Example lane change as used as input in the inverse optimal control algorithm.

The execution of this research is conducted with the support of "Siemens Digital Industries Software - NVH R&D engineering department" located in Leuven which made it possible to preserve the direct link with reality. Software was made available e.g. Simcenter Amesim and the possibility to validate the obtained algorithms with real driver data made it possible to make the results that could be obtained more significant.



## Chapter 2

# State of the art modelling of comfort

As discussed in the introduction the goal of the thesis is to learn and implement a method to capture personal experience of comfort in autonomous driving. This will be done by using an inverse optimal control approach where the weights are learned from demonstration. To be able to do this it is necessary that a literature study is done about how to define comfort in a vehicle and to gain information about inverse optimal control.

This chapter will give an overview of the literature that is available and will show how the thesis fits in earlier conducted research.

### 2.1 What are the parameters that define comfort during driving?

In the following US patent [5] the idea is to assess the amount of comfort by calculating a value for carsickness. This value is calculated by a weighted sum of the sway motion, surge motion and heave motion of the vehicle. These motions are being directly calculated from the lateral acceleration, fore-aft acceleration and the vertical acceleration of the vehicle.

In the paper 'Investigating ride comfort measures in autonomous cars' [7], it is explained that due to the introduction of autonomous vehicles there will be an other perception of comfort. Figure 2.1 indicates in blue the claimed traditional comfort factors and in red the new ones that also have to be taken into account in when driving in autonomous vehicles. Concretely this can be translated into the preference of smooth trajectories and low lateral motions when the roads are assumed to be sufficiently smooth. A hypotheses is that motion sickness will be more prominent in autonomous driving due to the loss of control. Is is also argued that the amount of travel time and the distance to an obstacle are naturally parameters that contribute to a comfortable feeling.



FIGURE 2.1: Overview of comfort parameters in autonomous vehicle with old parameters (blue) and new ones (red). (Source: [7])

In 'Analysis of Driving Style Preference in Automated Driving' [3] three studies are conducted in order to capture the definition of comfortable driving in autonomous vehicles. In the first study drivers drove manually with their own driving styles and this data was used in order to look for relevant metrics that could be used in defining distinct driver styles. It was found that these metrics are varying across the maneuver. This is a logical result because not in all maneuvers are all the comfort matrices equally dominant.

The results of this research is that accelerations play a key role in comfort but are not the only factor. [3]

The different comfort metrics found:

- lateral and longitudinal acceleration;
- lateral and longitudinal jerk;
- quickness of maneuver;
- headway distance;

Quickness of completing a lane change or an acceleration maneuver could respectively be modelled as:  $q_L = \frac{\int \frac{velocity \cdot dt}{\Delta distance}}{Time}$  and  $q_A = \frac{\int \frac{acceleration \cdot dt}{\Delta velocity}}{Time}$ .

Comfortable driving as being assessed in [3] can be summarized as driving with small accelerations and jerk to obtain sufficient smooth trajectories and keeping sufficient headway distance in order to have a feeling of control and safety. These results suggest that when an algorithm to assess comfort is suggested for autonomous

vehicles, a notion of the surrounding traffic should be inherently present. It was found that in order of the traffic density the driver is more tolerant towards less smooth driving behaviour e.g. to be able to insert in a busy lane. In this case higher comfort can be attained if the driver has a feeling of a fast responds of the vehicle translating in early peaks of acceleration. Vertical vibrations come not into scope when roads are assumed sufficiently smooth.

In a second study the main metrics that were found from study one are varied and combinations are rated by the use of a survey about the amount of comfort. "Out of this it followed that accelerations are playing a key factor." [3] For lane changes it could be concluded that the maneuvers with a small lateral acceleration and early perceivable onset were more comfortable. It is further also hypothesised that the location of the acceleration peaks and the amount of symmetry of the acceleration signal also plays a role when the amount of comfort is rated.

The relation between jerks and comfort is also confirmed by [8] where it is stated that: "Jerk has been shown to elicit a stronger influence on comfort than acceleration".

### 2.1.1 Conclusion

In order to find parameters of driving comfort that will further on in the thesis will be used as comfort criteria, a literature study is conducted which is mainly questionnaire based. The results are that in order to be able to assess comfort higher order kinematic variables as accelerations and jerks should be taken into account. These are important variables in order to obtain a smooth vehicle path and to give a continuous and natural feeling when driving. Also should the quickness of the maneuver and the feeling of safety of the driver be taken into account. As quoted by [19] states that low frequency lateral acceleration are the mean responsible for carsickness. In addition comfort assessment is influenced by the environment because in busier traffic there is a higher tolerance for less smooth trajectories. It was also found that the effect of a fast reaction of vehicle at the start of a maneuver is influencing the amount of comfort perceived in a positive way.

## 2.2 Inverse reinforcement learning

Because every human has its own driving style it is a cumbersome task to tune these parameters for each individual in order to model a personal perception of comfort. In [15] it is showed that manual tuned parameters will lead besides also to sub-optimal solutions in comparison when the parameters are learned. That is why it is chosen in this thesis to derive these parameters by demonstration of a human driver.

The goal of the learning algorithm explained in this thesis is to derive the parameters of different linearly combined comfort criteria combined in the costfunction  $J$  that when optimized as best as possible explain the observed data. As the match with the observed data gets better, the model as suggested by equation 2.1 is getting

closer to the inner comfort model of the individual driver. However it should be noted that the driver is taken unknowingly a lot of comfort criteria into account and they are not always linearly relating. Therefore the suggested comfort cost function consisting of a linear combination of comfort criteria will always be an approximation of the reality. Yet as discussed in [11] the results suggest that the magnitudes of the quantities that contribute to the comfort of the user are obtained.

$$J = \sum_{j=1}^N \theta_j \cdot f_j \quad (2.1)$$

In order to create a generative model that create vehicle paths  $r_i$  with equivalent kinematic characteristics as the path that was observed  $\tilde{r}_i$ , a feature-based inverse reinforcement learning is applied. [11, 1] With  $i \in [1...m]$  and  $m$  the amount of observed trajectories. A feature is encoding relevant kinematic properties and the difference between the demonstrated and calculated features says something about the similarity of the kinematic vehicle signals.

### 2.2.1 Background on learning algorithm

A feature values maps a trajectory onto a scalar and the higher the scalar value the more discomfort is experienced. An example of a that measures the amount of accelerations in a maneuver is given by equation 2.2.

$$f : \mathbf{r} \rightarrow f(\mathbf{r}) = \int_0^T \ddot{x}(t)^2 + \ddot{y}(t)^2 dt \quad (2.2)$$

The path that the centre of gravity of the vehicle is following can be represented by: 2.3.

$$\mathbf{r} : t \rightarrow \mathbf{r}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} \quad (2.3)$$

The driver is not a deterministic agent and is modelled by a stochastic distribution:  $p(\mathbf{r}|\boldsymbol{\theta})$  For certain weights  $\boldsymbol{\theta} \in \mathbb{R}^N$  a path  $\mathbf{r}$  is produced as being a sample of a stochastic distribution. The distribution that is chosen for this is the distribution of maximum entropy (equation: 2.4). [4]. This describes the data best because it is the distribution that is least biased. [11] The distribution with the highest entropy represents the given information best since it does not favour any particular outcome besides the observed constraints. [1]

$$p(\mathbf{r}|\boldsymbol{\theta}) = \exp(-\boldsymbol{\theta}^T \cdot \mathbf{F}(\mathbf{r})) \quad (2.4)$$

Equation 2.4 can be interpreted as a linear costfunction  $\boldsymbol{\theta}^T \mathbf{F}(\mathbf{r})$  where agents are exponentially more likely to select trajectories with lower cost. [11] The observed feature vector  $\tilde{\mathbf{F}} \in \mathbb{R}^N$  has on its entries the different feature values  $\tilde{f}_i$  with  $i \in [1...m]$ . The averaged observed feature vector is  $\tilde{\mathbf{F}}_{av} = \frac{1}{m} \sum_{j=1}^m \tilde{\mathbf{F}}_j$ .



In order to be able to explain the averaged observed feature vector, the weights  $\theta$  need to be found that match the expected features vector obtained from the distribution  $\mathbf{F}(r_{expected})$  with the averaged observed features vector  $\mathbf{F}_{obs}$ .  $r_{expected}$  defined as  $E(p(\mathbf{r}|\theta))$ . To make the expected features vector match the averaged observed features vector, gradient descent method can be used with as gradient  $\mathbf{F}_{obs} - \mathbf{F}(r_{expected})$ . Equation 2.5 summarizes the gradient descent method with  $\alpha$  the step size taken in the direction of descent.

$$\theta^{k+1} = \theta^k - \alpha \frac{\partial \mathbf{F}^k}{\partial \theta} \quad (2.5)$$

When  $\theta_{optimal}$  is found the gradient is minimized and the features vectors will match as closely as possible. "When the feature expectations match, guaranteeing that the learner performs equivalently to the agent's demonstrated behaviour regardless of the actual reward weights the agent is attempting to optimize" [1]

In order to calculate  $\mathbf{F}(r_{expected})$  a Hamiltonian Markov chain Monte Carlo stochastic distribution sampling approached is used in [10]. In [11] a simplified method and less calculation demanding approach is proposed where it is assumed that the expected path is the one that is been assessed as the most comfortable by the driver and is thus the path that is minimizing 2.1 for certain weights. In order to be able to match with the observed data it is hereby assumed that the demonstrations not only are samples of a stochastic distribution but are also generated by optimizing an inner cost function which is similar to 2.1. Equation 2.6 summarizes the approximation proposed by [11].

$$r_{expected} = \underset{\mathbf{r}}{argmax} p(\mathbf{r}|\theta) = \underset{\mathbf{r}}{argmin} \theta^T \cdot \mathbf{F}(\mathbf{r}) \quad (2.6a)$$

$$\frac{\partial \mathbf{F}}{\partial \theta} = \mathbf{F}_{obs} - \mathbf{F}(r_{expected}) \quad (2.6b)$$

It is checked in chapter 3 if this approach that is assuming that demonstrations are generated by minimizing a cost function also gives acceptable results when the this is violated.

### 2.2.2 RPROP algorithm

From [14] it is known that not every step size in the opposite direction of the gradient is leading to the convergence towards a minimum. When the step size is too small it will take a long time to convergence. However when it is chosen too big, cycling behaviour between limit points can occur. In order to avoid this kind of unwanted behaviour a line-search method is needed in order to change the step size taken during the course of the algorithm. For this the Resilient backpropagation algorithm (**RPROP**) [17] is used as it was first proposed by Martin Riedmiller and Heinrich Braun in 1993.

The main advantage when using RPROP is that the size of the gradient is not blurring the update value of the weights for the start of the next iteration in the gradient descent optimization method. The update value  $\Delta u$  is solely dependent of the sign of the current gradient and the sign of the gradient in the previous iteration. The main equations of the algorithm are given for each by:

$$\Delta u_i^t = \begin{cases} \eta^+ \cdot \Delta u_i^{t-1}, & \text{if } \frac{\partial f_i}{\partial \theta_i}^t \cdot \frac{\partial f_i}{\partial \theta_i}^{t-1} > 0 \\ \eta^- \cdot \Delta u_i^{t-1}, & \text{if } \frac{\partial f_i}{\partial \theta_i}^t \cdot \frac{\partial f_i}{\partial \theta_i}^{t-1} < 0 \\ \Delta u_i^{t-1}, & \text{if } \frac{\partial f_i}{\partial \theta_i}^t \cdot \frac{\partial f_i}{\partial \theta_i}^{t-1} = 0 \end{cases} \quad (2.7)$$

When the update value of the weight is determined it is applied in the direction of steepest descent which equals the opposite direction of the current gradient.

$$\Delta \theta_i^t = \begin{cases} -\Delta u_i^t, & \text{if } \frac{\partial f_i}{\partial \theta_i}^t > 0 \\ +\Delta u_i^t, & \text{if } \frac{\partial f_i}{\partial \theta_i}^t < 0 \\ 0, & \text{else} \end{cases} \quad (2.8)$$

Exception on 2.8:

$$\Delta \theta_i^t = -\Delta \theta_i^{t-1}, \text{ if } \frac{\partial f_i}{\partial \theta_i}^t \cdot \frac{\partial f_i}{\partial \theta_i}^{t-1} < 0 \quad (2.9)$$

$$0 < \eta^- < 1 < \eta^+ \quad (2.10)$$

and with  $\theta_i^{t+1} = \theta_i^t + \Delta \theta_i^t$ ,  $i \in \mathbb{N}_{[1 \dots N]}$  and  $t \in \mathbb{N}_{[1 \dots \tau]}$  the amount of iterations. Every time the partial derivative  $\frac{\partial f_i}{\partial \theta_i}^t$  of the corresponding weight  $\theta_i^t$  changes its sign, it is assumed that the last update was too big and the local minimum was passed. In 2.7 the step size is reduced and in order to go back the previous situation the update of the weight is done as 2.9. In order to not again decrease the update value when going back to the previous situation, in the next iteration  $\frac{\partial f_i}{\partial \theta_i}^{t-1}$  is set equal to zero. If the derivative retains its sign, the update-value is slightly increased in order to accelerate convergence in shallow regions." [17] Parameters chosen by the user are  $\Delta_0, \eta^+$  and  $\eta^-$ . In this thesis following values were chosen:  $\Delta_0 = 0.1$ ,  $\eta^+ = 1.2$  and  $\eta^- = 0.5$ .

## Conclusion

A background from literature on the inverse reinforcement learning algorithm used in this thesis was given. First inverse reinforcement learning was explained and how it could help to solve the research question how to learn comfort from observations. Afterwards a more detailed background on the formulation of the specific learning algorithm was given, which was complemented with a discussion on the algorithm used to update the weights being part of the gradient descent method to match expected feature values with observed ones.

## Chapter 3

# The Learning Algorithm

Bespreek het vehicle model. Lees features stukje na dat geschreven heb voor taak VS. X en Y variables zijn in longitudinal axis. Maakt gebruik van een complexer model  $\rightarrow$  kromming van het pad zit al inherent in de normal acceleration van het voertuig.  $\rightarrow$  versnellingen die ik beschouw zijn de totale versnellingen.

De gevonden trajectories vergelijken met een standaard lane change uit de literatuur  $\rightarrow$  zie *lanechange.in* voor foto.

Zeg dat eerst de approach was tested in ideale omstandigheden: zelfde voertuigmodel werd gebruikt en de assumptie dat de data is gegenereerd door de minimalizatie van een comfort kostfunctie werd aan voldaan.

Herinner de lezer nog even de structuur die gaat worden gevolgd. Learning, planning, tracking, validatie. Dit hoofdstuk zal over de learning gaan. Hoe is algorithm opgebouwd? Wrm wordt dit zo gedaan? Welke vehicle modellen wordt er gebruikt? Wrm mag men hier een simple vehicle mode gebruiken? Dit is gemachtigd omdat men hier de omgeving wil scannen voor een feasible pad  $\rightarrow$  dit wordt trager gedaan dan de tracking. (tracking zal gebruik maken van een meer complex model) Path planning ligt focus vooral op de omgeving. Goed refereren naar het rapport en VS rapport Wat zijn de assumpties die werden genomen?

Hoe zal de methode gevalideerd worden? Leg de twee methodes uit: code generatie en kijken of de wegings factoren terug gevonden kunnen worden? Mappen de feature values met de values van het geobserveerde pad?  $\rightarrow$  is het doel dat gevolgd probeert te worden haalbaar?

Ga hier niet meer te diep in op de entropie. Leg het hier meer intuïtief uit om de lezer niet te verwarren.

Vermeld afleiding van algorithm. Leg uit in Thesis hoe komt aan gradient die gebruikt. Zie papers: Ziebart et al and Kretzschmar et al.

Modeleer een andere bestuurder. Can try to reproduce a data set with a change of parameters which represents a different driver. Can check that the learned model is also different. Hiermee aantonen dat er ook echt andere wegingsfactoren worden gegenereerd en dat de specifieke driving characteristics worden meegenomen.

Ligt een tipje van de sluier op : hoe zal de data gegenereerd worden? Plot simulink

### 3. THE LEARNING ALGORITHM

---

model en duidt de blokken aan die zullen worden ingevuld. Hier gaat dieper in gegaan worden in de volgende hoofdstukken.

Maak een vermelding dat men het menselijke gedrag van het geleerde model kan nagaan met een Turing test.

Maak een plotje zoals paper Learning to Predict Trajectories of Cooperatively Navigating Agents → feature variance afwijking en average error. (zelfde plotjes als al de papers)

Schrijf een paragraaf over hoe de data gegenereerd wordt. → leg kort het gebruik van de verschillende vehicle modellen uit.

Schrijf een paragraaf over de theta update → zie RPROP methode → beschrijf wrm beter is dan andere methodes die gezien werden. Bespreek hoe de parameters werden gekozen.

Kan vermelding maken dat in deze thesis de features zijn gekozen met de hand → men kan proberen om de features ook te leren van data (Characterizing Driving Styles with Deep Learning)

## **3.1 The First Topic of this Chapter**



## Chapter 4

# Background in optimal control problems

This chapter gives some background information of the theory behind optimal control (OCP). After a global introduction and the discussion about the time discretization and shooting option, the chapter is being specified into model predictive control (MPC).

### 4.0.1 Optimal control problem (OCP)

An optimal control problem determines the desired inputs and corresponding state trajectories to change the system from an initial state to a desired final state in an optimal way while satisfying some input and state constraints [12].

$$\begin{aligned} \min_{\mathbf{q}(\cdot), \mathbf{u}(\cdot)} \quad & \int_0^T l(\mathbf{q}(t), \mathbf{u}(t)) dt + E(\mathbf{q}(T)) \\ \text{s.t.} \quad & \dot{\mathbf{q}}(t) = \mathbf{f}(\mathbf{q}(t), \mathbf{u}(t)) \\ & \mathbf{q}(0) = \mathbf{q}_0, \quad \mathbf{q}(T) = \mathbf{q}_T \\ & h(\mathbf{q}(t), \mathbf{u}(t)) \geq 0 \\ & \mathbf{q}(t) \in Q, \quad \mathbf{u}(t) \in U, \quad t \in [0, T] \end{aligned} \tag{4.1}$$

$\mathbf{q}$  is called contains all the states of the system and  $\mathbf{u}$  containing the controls. In the context of vehicle control states are often kinematic variables like positions and velocities of the centre of gravity of the vehicle.  $\mathbf{u}$  is containing the controls which are typically steerwheelangle and the amount of throttle which can be directly linked the amount of propulsion force. The objective function  $l$  of the optimization problem is integrated over the desired control horizon  $T$ . The objective function indicates what should be minimized and is a function of the different states and controls. The terminal cost is represented by  $E(\mathbf{q}(T))$  and can be needed to assure certain conditions of the system at the end of the control horizon.  $\dot{\mathbf{q}}(t)$  describes the dynamics of the system by an explicit ordinary differential equation. Furthermore there are also constraints possible on states and inputs, represented by  $h$ ,  $Q$  and

$U$  respectively. The results that come out of an OCP indicate which states will be visited by the system and which controls have to be applied in order to do this with respect to the constraints on an optimal manner. [13]

There is a difference between soft and hard constraints. Soft constraints are placed in the objective function  $l$ . If the constraint is better fulfilled a more optimal solution will be obtained. A hard constraint, represented by  $h$  in equation 4.1, is explicitly put in the constraints and defines the feasible solution set of the optimization [21].

#### 4.0.2 Time discretization

The optimal control problem (equation 4.1) is continuous in time which means that it has infinite dimensions. To be able to run the optimization problem on digital systems there is need for discretization. There are several ways to do this which are summarized in Figure 4.1.

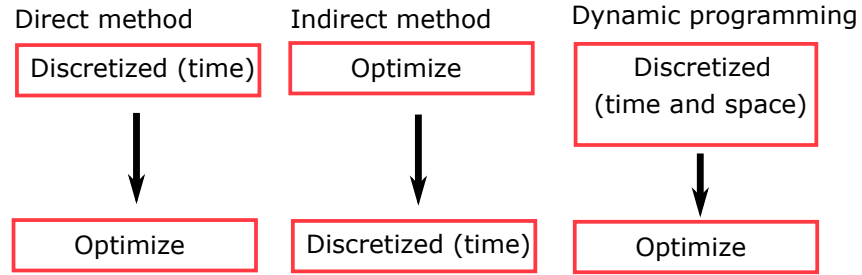


FIGURE 4.1: Overview of different discretization methods.

Since direct methods are best suited to solve practically relevant OCPs [12], this thesis is following the direct method. To implement the discretization of time when using the Direct method, a time shooting approach can be used.

#### Time shooting

A shooting approach makes use of a time grid. This means that time will be sampled and on every time instant the optimal control problem is assessed. Constraints will only be not violated on these time instants but no limits are set between different time samples. To bound the system to the constraints a high enough sampling rate is desired. [12] Two different shooting approaches exist:

1. Multiple shooting (MS)

During multiple shooting  $ns \in \mathbb{N}$  new states and  $nc \in \mathbb{N}$  new controls are defined on every new time sample and are taken as optimization variables. Because input changes are only allowed on the time samples this will often lead to a piece wise control input signal. This is indicated by the blue bars in Figure 4.2 (left). The red dots in Figure 4.2 (left) indicate the condition



of the states on the discrete time samples. In order to make the connection  $\mathbf{q}(k+1) = \mathbf{f}(\mathbf{q}(k), \mathbf{u}(k))$  from the previous state to the next, time integration is used. The constraints introduced in this way are called in literature 'path closing constraints' [9].

## 2. Single shooting (SS)

In the single shooting or sequential approach only the initial state and control points are optimization variables. This is achieved by replacing the state variable by the integration result from the previous state [9]. This approach is show in equation 4.2. Figure 4.2 (right) gives a visualization with the green dots indicating the result of one integration step from the previous state.

$$\begin{aligned} \mathbf{q}(1) &= \mathbf{f}(\mathbf{q}(0), \mathbf{u}(0)) \\ \mathbf{q}(2) &= \mathbf{f}(\mathbf{f}(\mathbf{q}(0), \mathbf{u}(0)), \mathbf{u}(1)) \\ \mathbf{q}(3) &= \mathbf{f}(\mathbf{f}(\mathbf{f}(\mathbf{q}(0), \mathbf{u}(0)), \mathbf{u}(1)), \mathbf{u}(2)) \\ &\dots \end{aligned} \tag{4.2}$$

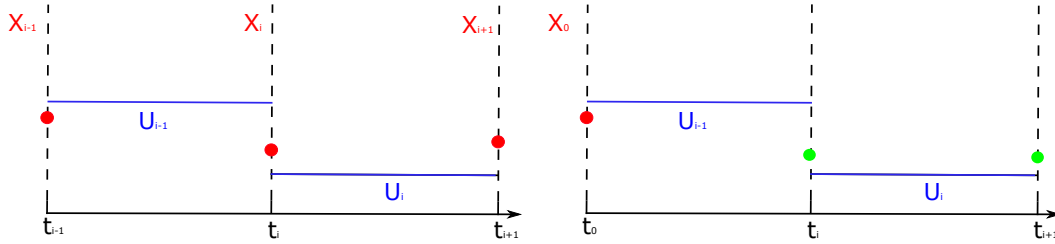


FIGURE 4.2: Schematic view of the time shooting approaches (left: multiple shooting; right: single shooting).

In this thesis a multiple shooting approach is used together with the use of a Runge-Kutta integration scheme. Runge-Kutta is an explicit integration scheme which has a higher calculation cost than a standard Euler scheme but is more reliable for non-linear systems and has a higher stability with respect to the chosen time-step [12].

Multiple shooting will lead to a larger Hessian of the objective function and a larger Jacobian of the constraints in comparison with the single shooting approach due to the fact that more optimization variables are introduced. But on the other hand, multiple shooting has a sparse Hessian which can be solved very effectively. Single shooting would instead often produce a smaller fully populated Hessian because of the very non-linear way that the states depend on the begin state and the different controls.

### 4.0.3 Model predictive control

MPC is already a mature approach in slow changing environments such as a chemical plant, but has more recently made also his breakthrough to the fast dynamic systems due to an increase of computational power and the implementation of new algorithms [12].

In order to be able to deal with model-plant mismatch and disturbances, MPC uses in this paper a moving control horizon. The time horizon is divided in discrete steps  $T_s$  and inputs are calculated over the finite prediction horizon  $N \cdot T_s$  by solving an OCP. The decision on the amount of samples taken to define the control horizon is based on a trade off between calculation effort and accuracy [18, 12]. Figure 4.3 depicts the solving of the OCP on time sample  $t + 1$ . In Figure 4.4 it can be seen how only the first control sample of the calculated control signal will be applied to the system and a new OCP is solved.

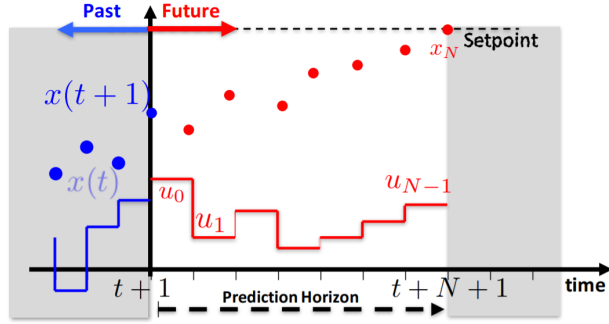


FIGURE 4.3: Visualization of the optimal control problem solved in one iteration of the MPC (Source: [13]).

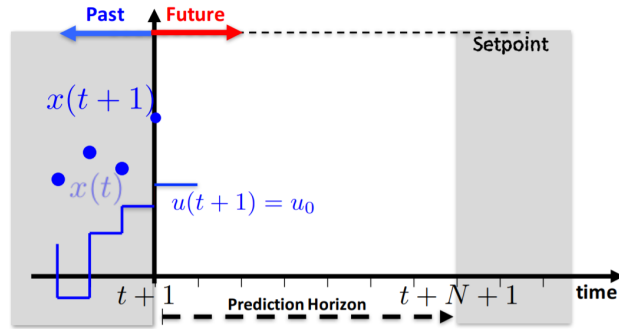


FIGURE 4.4: Visualization of the application of the first step of the calculated control signal during one iteration of the MPC (Source:[13]).

---


$$\begin{aligned}
& \min_{\mathbf{q}(\cdot), \mathbf{u}(\cdot)} \quad \sum_{k=0}^{N-1} l_k(\mathbf{q}_k, \mathbf{u}_k) + E(\mathbf{q}_N) \\
& \text{s.t.} \quad \dot{\mathbf{q}}_{k+1} = \sum_{k=0}^{N-1} \mathbf{f}(\mathbf{q}_k, \mathbf{u}_k) \\
& \quad \mathbf{q}_0 = \mathbf{q}_{measured} \\
& \quad h(\mathbf{q}_k, \mathbf{u}_k) \geq 0 \\
& \quad \mathbf{q}_k \in \mathcal{Q}, \quad \mathbf{u}_k \in \mathcal{U}, \quad N \in \mathbb{N}
\end{aligned} \tag{4.3}$$

Equation 4.3 is a representation of the solved discrete system OCP during one iteration of the MPC. It is a discretized version of equation 4.1. The Runge-Kutta integration is embedded in  $\mathbf{f}$ . The hard constraints are represented by  $h$ . It is worth noticing that the constraints can be violated in-between the different time sample points.

MPC has no direct feedback loop, but through the iterative way of solving the OCPs it can still deal with model mismatch or a changing environment. The downside of this approach is that it requires a bigger computational load, which makes efficiently written software a necessity.



## Chapter 5

# Path Planning MPC

MPC → path planning met het gevonden model.

Ga hier volledig in op wat MPC is. Valideer de MPC code door na te gaan wat de invloed is bij het variëren van de gevonden parameters. Hoeveel zal het verschillen?

Ga in op het gebruikte point model/non-linear model en bespreek de gelijkenissen en de verschillen van het learning algorithm. (zie ook notes VS en verbeteringen NAETS)



## Chapter 6

# Path tracking MPC

MPC → path tracking met non lin model. Bespreek non lin model Bespreek resultaten. Valideer de resultaten. Wat gebeurt er als de parameters iets anders worden geschat? Zeker de parameters voor de banden zijn moeilijk om te schatten. Bespreek de tekort komingen van de het model en valideer het model hoe de referentie wordt getrackt met andere parameter values. hoe robuust is de mpc? (Zei paper VS)

Hier kan men praten over de implementatie in het simulink model. Het zou goed zijn om het ACADO model te vervangen door een tracking MPC die in CasADi geschreven is. (reference path to follow wordt niet geupdated maar blijft constant in model → dit is niet hoe het werkt in de realiteit.) Kan als echt tegoei wilt doen ook inladen in de template die gekregen heb van Flavia → PID vervangen door MPC en path planner. Een simulink model is nodig om er het 15 dof vehicle model in te kunnen verwerken.

### 6.1 The First Topic of this Chapter

#### 6.1.1 Item 1

Sub-item 1

Sub-item 2

#### 6.1.2 Item 2

### 6.2 The Second Topic

### 6.3 Conclusion





## Chapter 7

# Validatie

Bespreek de validatie van de methode. Implementeer simulaties in prescan. Bespreek de verschillende software tools bij Siemens → Amesim, simulink, prescan. Hoe werken ze samen en hoe wordt de validatie precies gedaan? Wat zijn de resultaten? Install amesim and write a chapter about how the dataset is generated. How is the amesim model defined etc.

### 7.1 The First Topic of this Chapter

#### 7.1.1 Item 1

Sub-item 1

Sub-item 2

#### 7.1.2 Item 2

### 7.2 The Second Topic

### 7.3 Conclusion



## Chapter 8

# Conclusion

The final chapter contains the overall conclusion. It also contains suggestions for future work and industrial applications.



# Appendices



# Appendix A

## The First Appendix

Appendices hold useful data which is not essential to understand the work done in the master's thesis. An example is a (program) source. An appendix can also have sections as well as figures and references[?].





## Appendix B

# The Last Appendix

Appendices are numbered with letters, but the sections and subsections use arabic numerals, as can be seen below.



# Bibliography

- [1] P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. *Proceedings, Twenty-First International Conference on Machine Learning, ICML 2004*, pages 1–8, 2004.
- [2] I. Bae, J. Moon, and J. Seo. Toward a comfortable driving experience for a self-driving shuttle bus. *Electronics (Switzerland)*, 8(9):1–13, 2019.
- [3] H. Bellem. *Comfort in Automated Driving : Analysis of Driving Style Preference in Automated Driving*. PhD thesis, 2018.
- [4] Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell and A. K. Dey. Maximum Entropy Inverse Reinforcement Learning Brian. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, page 6, 2008.
- [5] L. Daniel. Method and system for determining and dynamically updating a route and driving style for passenger comfort - US Patent, 2018.
- [6] E. David. Will autonomous vehicles be safe to use? | Cybersecurity & Technology News | Secure Futures | Kaspersky.
- [7] M. Elbanhawi, M. Simic, and R. Jazar. In the Passenger Seat: Investigating Ride Comfort Measures in Autonomous Cars. *IEEE Intelligent Transportation Systems Magazine*, 7(3):4–17, 2015.
- [8] C. Gianna, S. Heimbrand, and M. Gresty. Thresholds for detection of motion direction during passive lateral whole-body acceleration in normal subjects and patients with bilateral loss of labyrinthine function. *Brain Research Bulletin*, 40(5-6):443–447, 1996.
- [9] J. Gillis. Ya Coda course presentation, 2019.
- [10] H. Kretzschmar, M. Kuderer, and W. Burgard. Learning to predict trajectories of cooperatively navigating agents. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 4015–4020, 2014.
- [11] M. Kuderer, S. Gulati, and W. Burgard. Learning driving styles for autonomous vehicles from demonstration. *Proceedings - IEEE International Conference on Robotics and Automation*, 2015-June(June):2641–2646, 2015.

- [12] T. Mercy. *Spline-Based Motion Planning for Autonomous Mechatronic Systems*. PhD thesis, 2018.
- [13] P. Patrinos. Model Predictive Control - Lecture Notes, 2019.
- [14] P. Patrinos. Optimization - Lecture notes, 2019.
- [15] V. Powers, C., Mellinger, D., Kushleyev, A., Kothmann, B., Kumar. Experimental Robotics. *ISER, June*, 88(287513):515–529, 2013.
- [16] Prof. Amnon Shashua. Experience Counts, Particularly in Safety-Critical Areas | Intel Newsroom, mar 2018.
- [17] M. Riedmiller and H. Braun. Direct adaptive method for faster backpropagation learning: The RPROP algorithm. *1993 IEEE International Conference on Neural Networks*, pages 586–591, 1993.
- [18] Q. N. Tong Duy Son. Safety-Critical Control for Non-affine Nonlinear Systems with Application on Autonomous Vehicle. (August):7, 2019.
- [19] M. Turner and M. J. Griffin. Motion sickness in public road transport: The effect of driver, route and vehicle. *Ergonomics*, 42(12):1646–1664, 1999.
- [20] University of Warwick. Do passengers prefer autonomous vehicles driven like machines or like humans?, 2019.
- [21] K. Yankov. *Potential Field Based Model Predictive Control for Autonomous Vehicle Motion Planning and Control*. PhD thesis.
- [22] Yusof N.B.M. *Comfort in Autonomous Car : Mitigating Motion Sickness by Enhancing Situation Awareness through Haptic Displays* Nidzamuddin Md . Yusof. PhD thesis, Eindhoven, 2019.