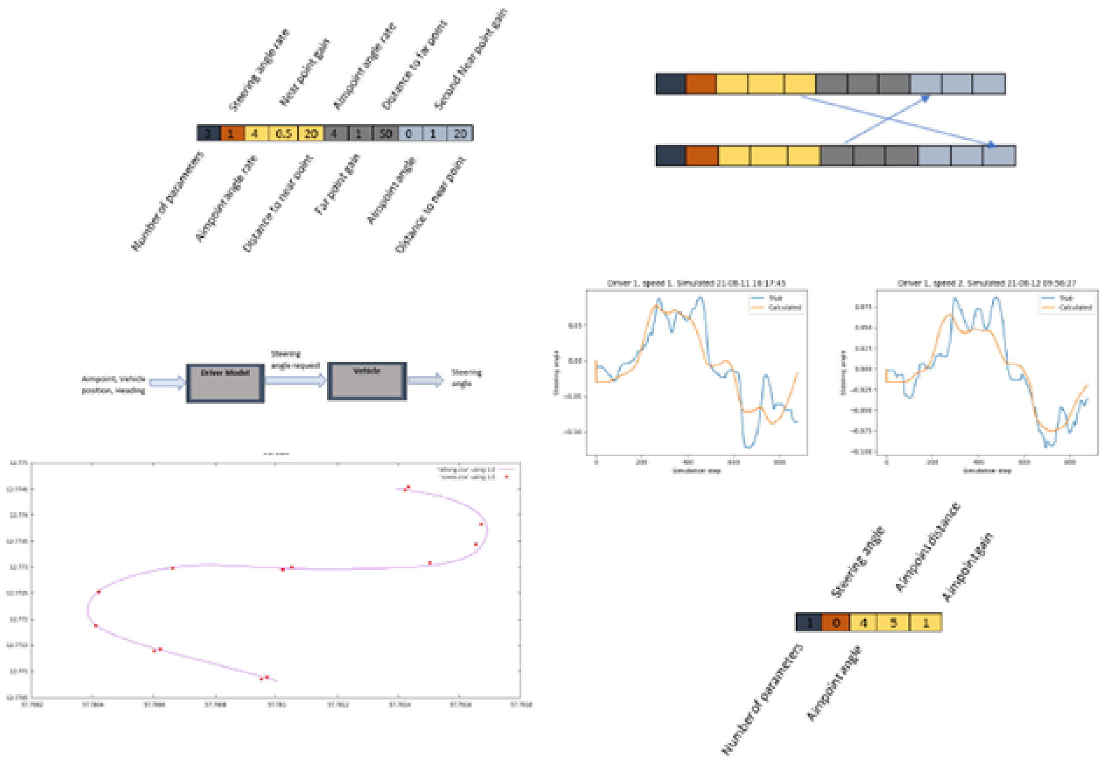
CHALMERS
UNIVERSITY OF TECHNOLOGY



Genetically generated bionic driver models for autonomous road vehicles

Master's thesis in Automotive Engineering and Data Science & AI

SHISHIR GURUSHANTHAPPA

OLLE LINDGREN

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2021
www.chalmers.se

Master's thesis 2021

Genetically generated bionic driver models for autonomous road vehicles

SHISHIR GURUSHANTHAPPA AND OLLE LINDGREN

Department of Mechanics and Maritime Sciences
Division of Vehicle Engineering and Autonomous Systems
Chalmers University of Technology
Gothenburg, Sweden 2021

Genetically generated bionic driver models for autonomous road vehicles
SHISHIR GURUSHANTHAPPA
OLLE LINDGREN

Supervisor: Ola Benderius, Department of Mechanics and Maritime Sciences
Examiner: Ola Benderius, Department of Mechanics and Maritime Sciences

Master's Thesis 2021:80
Department of Mechanics and Maritime Sciences
Division of Vehicle Engineering and Autonomous Systems
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Printed by Chalmers Digitaltryck
Gothenburg, Sweden 2021

Abstract

For autonomous road vehicles, control is often divided into longitudinal and lateral control. This thesis focuses on lateral control driver models derived from a cognitive perspective. A genetic algorithm is used to generate driver models expressed in a domain-specific language. The project focuses on isolating perceptual cues. The objective function for the genetic algorithm is computed as the difference between the estimated steering angles and the observed steering angles in the vehicle. The recordings were captured from a Volvo XC90 driving a single scenario with an S-shaped test track at different speeds, and with different drivers.

The resulting driver models are within 1-2 degrees of the recorded steering angles, and more significantly, the DSL sentences are very similar regardless of driver or speed, and stable between different runs. The project's results show that the implementation of the genetically generated driver models is possible for lateral control. This genetic algorithm serves as a platform for the future inclusion of external factors affecting the dynamics of the vehicle. The identified model and parameters can be tested for representing a real-world driving case.

Keywords: autonomous driving, domain-specific language, genetic programming, genetic algorithm, driver models, stochastic optimization, autonomous road vehicles.

Acknowledgements

I would like first to thank my mentor and supervisor Ola Benderius for his guidance and expertise and the discussions with him sparked new ideas and new areas could be explored during this time.

I would also take this opportunity to thank all my Chalmers Formula Student Driverless teammates and people working at Revere for their support. I want to express my gratitude to my family for their wholehearted support and especially my mother during this period. Finally, I thank all the people who have been a part in this wonderful journey of mine during my years at Chalmers.

Shishir Gurushanthappa, Gothenburg, September 2021

It is impossible to overstate how much I've learned over the past year, working with Chalmers Formula Student Driverless and Revere, and this thesis report doesn't scratch the surface of what that project entailed. However, I would still like to take this opportunity to thank not only those who made this thesis report possible, but also the key people who enabled our journey with Chalmers Formula Student Driverless.

So, in no particular order, I would like to thank Ola Benderius, Fredrik von Corswant, Stefan Lundberg, Mikael von Redlich, Diego Jauregui, Anthony Kalcic, and Lilian, Leo, Sam, Shishir, Gabriel and Sigrid from the Driverless team. You were all incredibly supportive throughout the project, and I will always be grateful for your participation in making the project possible.

Olle Lindgren, Gothenburg, September 2021

Contents

List of Figures

List of Tables

1
Introduction

Driver models are mathematical models that replicate key functions of a driver like controlling, monitoring and stabilizing a vehicle [1], and can be classified into three perspectives: control perspective, behaviour perspective and cognitive perspective. The control perspective is analogous to a closed-loop control system with the driver as the controller and the vehicle as the plant. The behaviour perspective is based on observation or hypothesized driver behaviour. Finally, the cognitive perspective uses bio-inspired methods for navigation, for example the vision and retinal flow of a driver's eyes [2].

This report considers driver models based on the cognitive perspective and heuristics approach. The most common approach for designing these types of driver models is by selecting a specific scenario, and designing a driver model that performs well in that.

Driver models with steering constrained trajectories often focus on road-edges to define the inputs, and since the approach is for a particular scenario, the lane boundaries actually define the steering task requirements by considering just the visual direction. However, there is a large body of literature highlighting the importance of perceptual cues information for the successful control of locomotion. The visual cues play a major role especially at higher speeds and also while manoeuvring vehicle. The perceptual cues in terms of the angular flow rate of objects for example could be considered for better control of the vehicle.

By having the perceptual cues also as inputs, the driver models can make large corrections of errors to compensate for the rapid error growth at each moment. In conditions where the speed of the vehicle is increasing, a feedback control strategy may result in oscillatory trajectories and convulsive steering.

Once the models are created, computer simulations can be done to study the system driver dynamics. Using simulations like these, the potential safety benefits of active safety systems and autonomous vehicles can also be studied, and the development of technology used for active safety systems and autonomous road vehicles may ultimately be furthered.

1.1 Scope and contributions

The aim of this thesis is to determine what type of information the drivers require while driving in a particular S-shaped track at AstaZero test track facility and create driver models. The created driver models are compared and the best driver model is deemed to be the best fit for validation in the real world driving scenario.

The thesis is trying to isolate perceptual cues by using *genetic programming* (GP) along with a *domain-specific language* (DSL) to find driver models that replicate the steering input of a human driver. This is done for different drivers and speeds, and the resulting DSL sentences can easily be interpreted as minimum parts that add up to functioning lateral control. By generating driver models for different drivers and speeds separately, something can also be said for how general the generated parameters are.

With this approach, it is possible to generate arbitrary linear combinations of a number of common driver models that are functions of preview points along a calculated target trajectory ahead of the car. Models such as the two-point model suggested by Salvucci and Gray, the racing driver model proposed by Sharp*et al.*, and the aim point model proposed by Benderius, can all be described as special cases of this approach.

1.2 Research questions

Every year, many accidents are attributed to the human error. At the same time, it is difficult to fully replace human drivers with an artificial one, so humans are still generally the safest drivers. It is however possible to design domain-specific systems that detect common examples of bad human driving and take control of the vehicle when those occur.

To better understand human driving, and to contribute to the development of artificial drivers, we would therefore like to answer the following research questions:

1. What information is a driver using when carrying out the specified S-shaped steering manoeuvre, e.g. what is the minimum set of driver model components that best generate the desired steering output?

2. How generic are driver model parameters between different drivers and different longitudinal speeds?

1.3 Limitations

This project will focus on lateral control. Any longitudinal control will be applied unchanged from the collected data, where only steering is being reproduced from the model. The constraints will be based on the type of data collection itself for the driver models.

One of the limitations for designing the driver models obtained from the test track experiments, in particular, itself is that the behaviour of the drivers driving the car at different constant speeds may vary for the given track and hence the vehicle positions as well if compared for the exact same situation in terms of manoeuvring.

2
Background

A considerable amount of research has been done on how human drivers control vehicles, using experiments on test tracks, in simulators, and with naturalistic data collection. In simulations, driver models have been shown to accurately replicate how human drivers control vehicles, with the advantage of being a cost-effective and endlessly repeatable tool to conduct research. This has led to the research and development of *advanced driver assistance systems* (ADAS) and even fully autonomous vehicles, although currently only tightly controlled environments.

2.1 Recent driver models for lateral control

In the early 2000s, extensive research of driver modelling paved way for ADAS to reach higher levels of semi-autonomous and autonomous driving, and models for lateral motion control were developed using control, behaviour and cognitive approaches.

In 1995, Land and Horwood proposed a method that measures steering accuracy as the ratio of the standard deviation of the angle between the vehicle heading and road centerline with the whole road visible to the standard deviation with one or two segments visible with three conditions of the segments move-near, move-far and move-both [4].

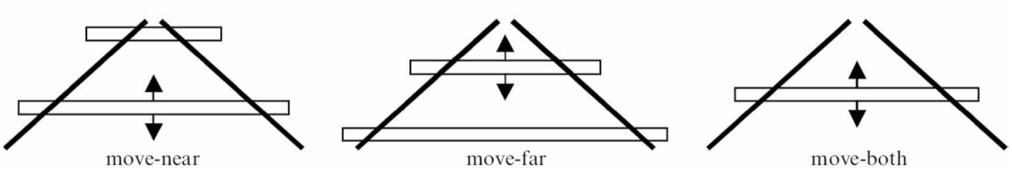


Figure 2.1: Illustration of Land and Horwood conditions for empirical study for visual field [3].

In 2003, Salvucci and Gray used a cognitive approach for lateral vehicle control, with a two-point control method using perceptual inputs. The approach uses a 'near point' and a 'far point', and the visual direction to each point. A lateral driver model is proposed that incorporates the near point to maintain a central lane position, and the far point to interpret for the impending roadway curvature [3].

In 2006, Sharp et al proposed a heuristic driver model to minimize the lateral deviation of the car from an optimal path ahead. The model uses multiple preview points, and for each preview point an optical lever is computed. This optical lever distance is a function of the vehicle's longitudinal velocity, and the steering angle response is calculated as a linear combination of the levers [5].

In 2014, Ola Benderius proposed a driver model that aimed to explain human driving behaviour using reaching theory. The model is satisfysing rather than optimising, and was developed to reduce the parameter redundancy shown by the Salvucci and Gray driver model. The aimpoint correction model uses a single preview point ahead of the car, and the steering input is made to be proportional to the lateral drift of the preview point from the drivers point of view [6].

In 2005, the DARPA Grand Challenge was held for autonomous driving. The driver model used in the winning car was based on the conversion of global coordinates given by the competition to the local coordinates in which the lateral position corresponds to the offset from the centre of the road and the longitudinal position to the distance travelled. This is one of the best examples of an autonomous vehicle that is capable of fulfilling the basic layers necessary for full-scale autonomous vehicles like selecting the destination. mission, global path planning, local path planning and control [7].

In 2012, Markkula, Gustav Markkula, Ola Benderius, Krister Wolff and Mattias Wahde in their review of near-collision recent driver models, concluded that most of the considered driver models are good enough to give the expected predictions of steering angle and speed of the vehicle. The authors emphasized that the validation of the models on relevant human data has rarely been achieved to the same level of detail, and that comparisons among driver models should be made more frequently, since the best driver model may be heavily scenario dependent [8].

In 2014, Markkula, Benderius and Wahde compared different driver models for driver steering behaviour in collision avoidance and vehicle stabilization and validated the models of lateral control. The authors used a *genetic algorithm* (GA) to fit parameters for each model, and then evaluated the fitted driver models on ground truth data. This approach provided driver model output very close to the recorded human driving control [9].

In 2020, Jauregui and Vukman, in their study of different driver models used a domain-specific language to generate driver models as DSL sentences. The study successfully evaluated the possibility to use GP algorithms to evaluate and compare not only the existing driver models but also generate new driver models [10].

In 2019, a DSL for autonomous driving scenario representation was put forth by Rodrigo et al. The study evaluated the levels of safety of autonomous vehicles. The DSL included test scenarios for different traffic situations and used this language as a tool to find suitable scenarios to simulate as a step in the validation process for autonomous driving software [12].

2.2 Driver models for lateral control

2.2.1 Sharp et al model

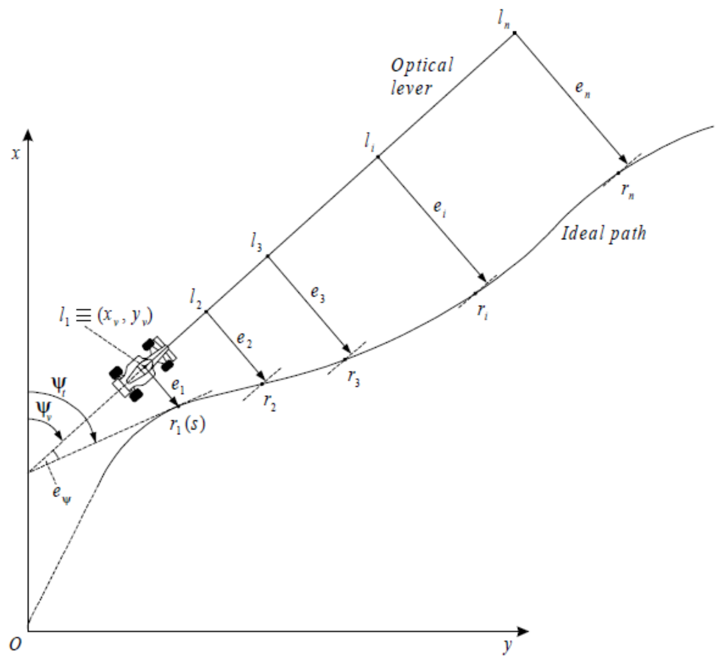


Figure 2.2: Schematic illustration of Sharp et al driver model [5].

The Sharp et al model is a mathematical model where the driver steering control is defined in terms of preview time, the number of preview points and their positions along the optical lever as shown in the above figure. The optical lever preview distance L is calculated as the product of vehicle velocity and the preview time from the vehicle centre. The mathematical model is given by

$$\delta = K_\psi e_\psi + K_1 e_1 + \sum_{i=2}^{n} K_i e_i \tag{2.1}$$

where δ is the steering angle output, K_ψ, K_1, and K_p are set of gains or free model parameters, n is the number of preview points, their spacing along the optical lever, e_i is the previewed path deviations and K_i is the exponentially decreasing profile for the preview gains (with $2 \leq \text{i} \leq \text{n}$) as from the literature. So in conclusion the steering angle is calculated as the weighted sum of current and previewed path deviations along a forward optical lever, extending a preview time ahead, and the current deviation. The model is trying to correct the heading of the vehicle by minimizing the lateral offset sensing the upcoming curvatures.

2.2.2 Salvucci and Gray driver model

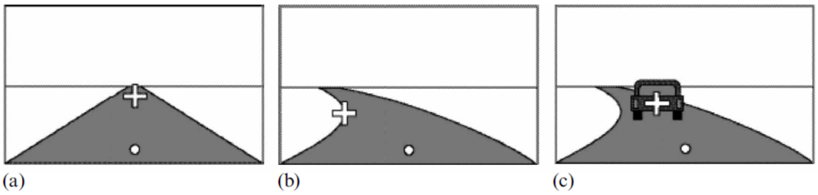


Figure 2.3: Schematic illustration of Salvucci and Gray driver model. Near and far points for three scenarios: (a) straight roadway with a vanishing point, (b) curved roadway with a tangent point, and (c) presence of a lead car [3].

The Salvucci and Gray model includes a two-point model approach, a near point and a far point for controlling the vehicle. The driver model emphasizes more on the heuristic approach for driver modelling by inculcating what type of information or specific sources of perceptually available information, often referred to as perceptual cues are used in driving control. The mathematical model is given by

$$\dot{\delta} = K_n\dot{\theta}_n + K_f\dot{\theta}_f + K_i\theta_n \tag{2.2}$$

where $\dot{\delta}$ is the rate of change of steering angle rather than just the steering angle. The first term in the equation is split into two terms, one that represents the contribution of the change in far-point visual direction.

2.2.3 Benderius driver model

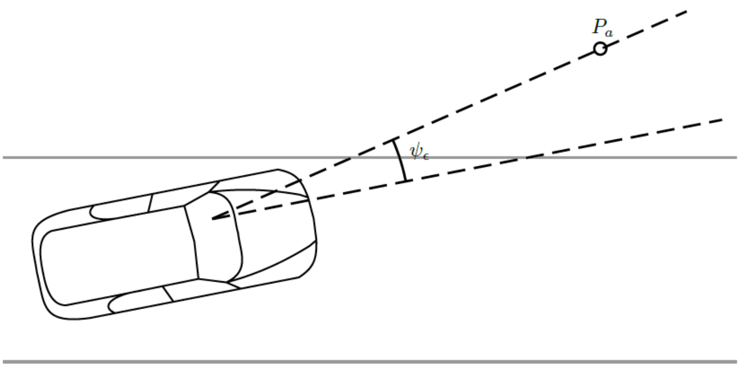


Figure 2.4: Schematic illustration of Benderius driver model. A single lane change is performed where the driver fixates at the aim point P_a. In the model presented here, the angle is used by the driver as the only heuristic for lateral control [6].

The Benderius driver model is a aimpoint correction method taking the cognitive approach and clearly separating the heuristics and control. The aimpoint is a fixation point at a long headway during normal driving in a straight road and positioned at a short headway in curvatures or other overtaking and several scenarios where the fixation point headway can be reduced. The aimpoint is the point where the vehicle wants to head and orient towards. This is done with the use of error between the aimpoint and the current heading of the vehicle as from the optical instrument's perspective is the difference between the current and desired headings. One of the important things in this driver model is that though the near point is implicitly defined, the information can be used to establish an egocentric reference frame. The reference frame is used for longitudinal and lateral directions. The mathematical model is given by

$$P_a = (x_a, y_a) \tag{2.3}$$

$$\psi_e = tan^{-1}\left(\frac{y_a - y(t)}{x_a}\right) - \psi_h(t) \tag{2.4}$$

$$\delta(t) = k_s(t)\psi_\epsilon(t) \tag{2.5}$$

where, P_a is the aim point in the current egocentric reference frame, ψ_ϵ is the aimpoint angle error, y is the driver lateral offset, and h the vehicle and the driver heading in this case the optical instrument or tracker, k_s is the gain, and δ the steering wheel angle from the literature. The above equations aim at correcting the fixation point. The driver-predicted steering wheel angle or distance eliminates the aim heading error.

3
Methods

A set of possible driver models were defined as words in a domain-specific language, where a set of words formed a sentence. Such DSL sentences were used as driver models, where the steering input was simply the sum of the steering inputs of the individual words in the sentences. Using a genetic algorithm, random DSL sentences were generated, and using open-loop simulation of recorded data logs, the steering input from these logs were compared to the calculated steering input from the DSL sentences, and for every DSL sentence an objective function was computed as the mean square error between these two. This was repeated until convergence, for every driver and every speed. The software required for the simulations, genetic algorithm and DSL sentence generation was written in C++ and deployed as a docker container.

3.1 Domain Specific Language

A domain specific language is a descriptive language where the focus is on a particular domain and representing the parameters and individuals by notations and text. A DSL is a set of sentences that are formed by letters and governed by user-defined rules, by involving grammar or arrangement of letters in a finite set, for example to distinguish one parameter from another.

In this case, the DSL was a language focused on the domain of driver models for lateral control for autonomous vehicles. A DSL sentence represents a driver model, and includes its parameters.

To transform a DSL sentence into a driver model, an initial word that represents either steering angle or a steering angle rate was used. So the important step was to identify whether it was a steering angle or a steering angle rate that was required as an output. In conclusion, a DSL sentence was the sum of one word representing angle or rate and the sum of all stimuli.

Here, the DSL sentence initialized the type of driver model based on the number of words. The first part in the DSL sentence represented the number of components or stimuli involved in the driver model, the second part in the sentence represented the type of output and the third part was a combination of characters of stimuli which were a combination of types of word, gain and distance.

To summarize:

- The first part represented the number of parameters associated with the driver model. For example, the Salvucci and Gray driver model used 3.
- The second part represented the type of output, and was either steering angle or steering angle rate. For Salvucci and Gray, it was steering angle rate.
- The third part was the combination of parameters related to the driver model. In Salvucci and Gray, it was the first combination of aimpoint angle rate, distance to far point and far-point gain, the second combination was the aimpoint angle rate, distance to near point and near point gain and finally the third combination was aimpoint angle, distance to near point and second near point gain.

The type of word of driver model stimuli and the parameters associated with them were:

1. The first part of the word included:
   - Aimpoint angle
   - Aimpoint angle rate
   - Optical lever
   - Lateral offset
   - Heading error
2. The second part of the word included:
   - Gains associated with the first part of the word
3. The third part of the word included:
   - The distances related to the first part of the word.

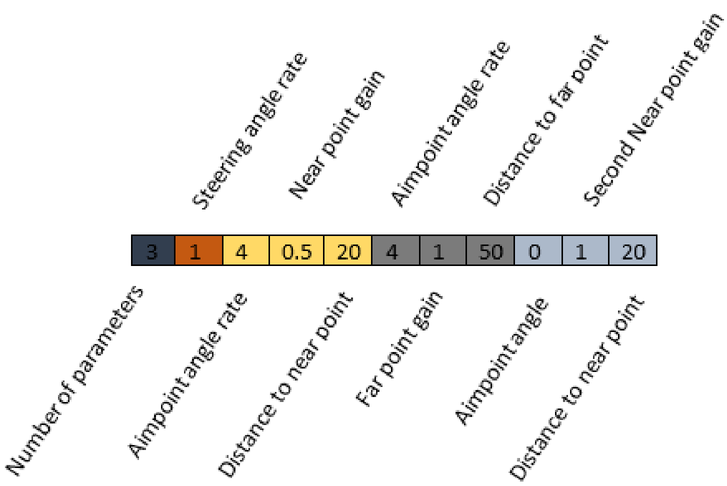


Figure 3.1: Illustration of a DSL sentence for the Salvucci and Gray driver model.

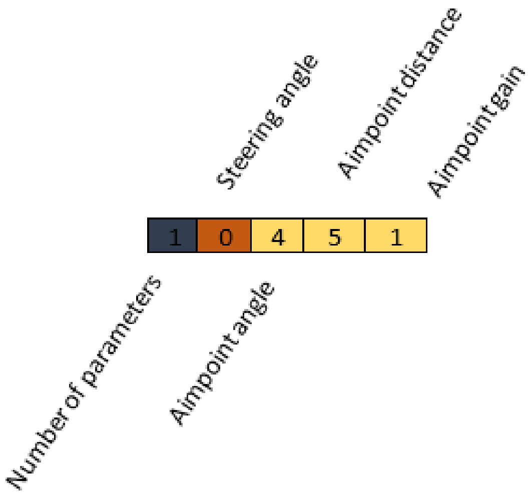


Figure 3.2: Illustration of a DSL sentence for the Benderius driver model.

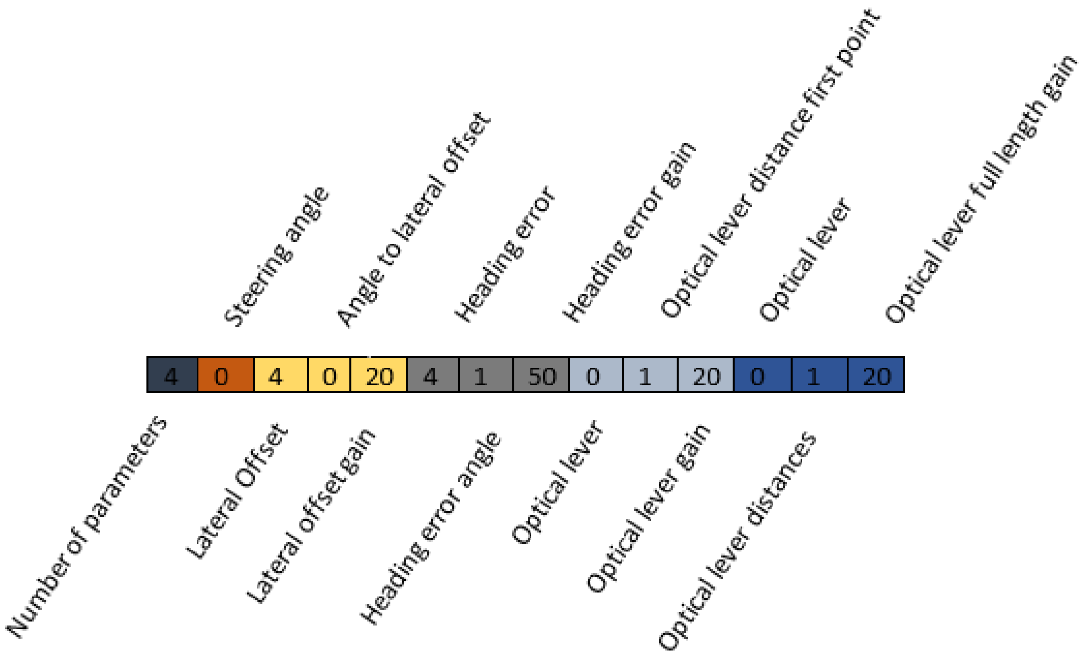


Figure 3.3: Illustration of a DSL sentence for the Sharp et al driver model.

3.2 Genetic programming

The genetic algorithms are part of evolutionary algorithms which are search and optimization techniques, inspired biologically from the theory of evolution where the concept of population, gradual and hereditary changes, generations and fitness are synonymous in the GP. The solutions obtained in GP tend to be adaptive unlike classical optimization techniques which are deterministic. GP is often suitable when the search space is complex or unstructured.

The GA initialization generated a population of individuals with randomized values, and the fitness function was evaluated for each individual in the population. The best few individuals were selected using tournament selection, and traditional crossover and mutation were used to create new individuals based on these. Elitism was also used, so that the best 10 individuals in a particular generation would always be carried over to the next generation. The references for GA in this thesis were taken from M. Wahde's book on the biological optimization methods an introduction [13].

3.2.1 Genetic operators

The fitness function was calculated using open loop simulation. For every individual, the selected scenario was simulated and the steering input of the individual driver model was recorded. These recorded numbers were then compared to the ground truth numbers as recorded in the vehicle logs, and the fitness function was the mean square error between these two:

$$Z = N^{-1}\Sigma_{i=1}^{N}(\delta_{gti} - \hat{\delta}_i)^2$$

where δ_{gt} was the recorded trajectory steering angle, $\hat{\delta}$ was the estimated steering angle, and N was the number of discrete simulation steps.

Selection was performed with tournament selection and the probability of selection was set to 0.5. This was done so that not only the most fit individual and derivatives thereof would dominate the population, but to also explore different variants with fitness values that may be lower. In tournament selection, the individuals in the population are iterated over from best to worst fitness, and with some set probability, each individual is selected.

Crossover was also performed at each iteration. During crossover, two individuals were first selected. For both individuals, a crossover point somewhere along their respective DSL sentences was selected, and two new individuals would be created with the first half of the first individual and the second half of the second individual, and vice versa.

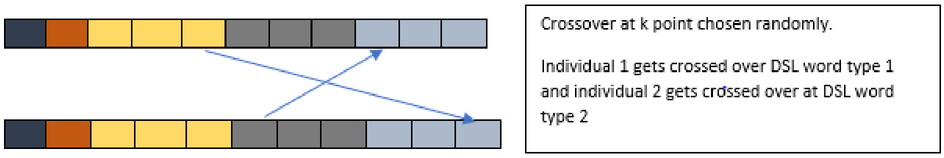


Figure 3.4: Schematic illustration of crossover for the considered randomly selected individuals.

Mutation was done with the crossed over individuals, with a probability of 0.05. In general, mutation negatively impacts fitness, but given a good selection process and enough time, mutation allows an individual to reach a local optimum almost certainly. During mutation, a small random number drawn from a normal distribution would have been added to the existing DSL words.

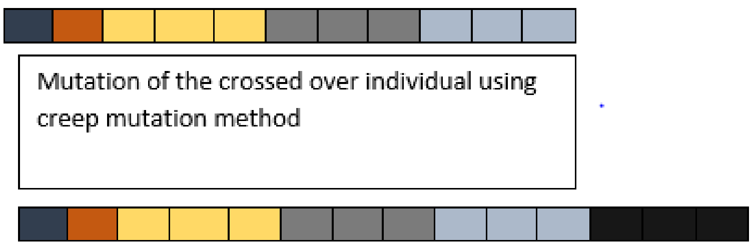


Figure 3.5: Schematic illustration of mutation of the crossed over offspring individual.

Elitism was used to ensure that the best individuals did not get lost. The number of individuals in the elite pool did not seem to matter, but 1 was used.

3.3 Recorded data and platform

Data was collected from a Volvo XC90 test car named Snowfox and operated by Chalmers. The car was driven by three different drivers at six different speeds at the AstaZero test track, and used on-road data logging. The data being recorded or collected by the data loggers was a means of creating naturalistic data but at a lesser scale. Some 40 different data points were recorded including GPS positioning, speed, acceleration, steering angle, IMU readings and many other values. The collected data was stored in different CSV files and interpolated for use in this report.

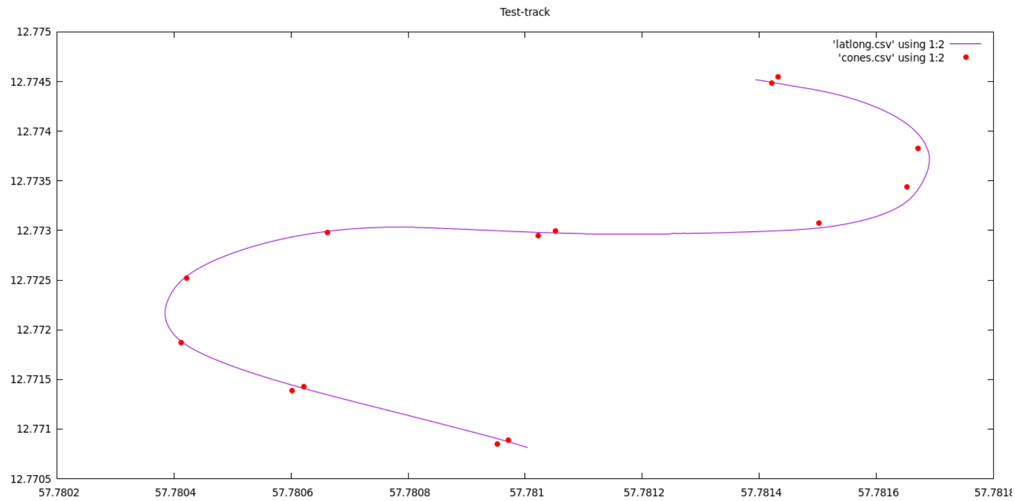


Figure 3.6: An example of GPS path taken by one of the drivers during a trial in AstaZero test-track facility.

In order to create ground truth data, the WGS84 GPS trajectories of different drivers at different speeds were used, and these GPS coordinates were converted into a local X/Y reference plane, and used to construct a preview path along which preview points were placed according to the DSL words. A schematic illustration of this process is available in Figure 3.8.

Figure 3.7: Snowfox, the Volvo XC90 test car used for the data collection [14].

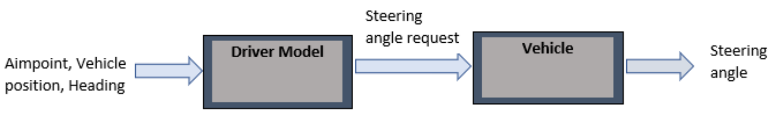


Figure 3.8: Schematic illustration of open loop driver model estimating the steering angle required.

The first step was to parse the recorded data. During this process, two trials were discarded due to early stops. The second step was to convert the global path to a local path that was from WGS84 geodetic readings to Cartesian coordinates for the simulation. The next step was to consider these converted recordings and make sense in terms of vehicle position, heading error, lateral deviation, preview distance for calculation of aimpoint or preview points. Then, the initialization of the driver model in the form of DSL sentences was done. These DSL sentences were individuals which had specific parameters for the considered driver models under study. Then these individuals were evaluated with fitness functions to determine the fitness values and then crossed over and mutated in consecutive generations to produce a steering wheel angle required by the vehicle according to the respective driver models. These obtained steering wheel angles were compared with the known human steering input.

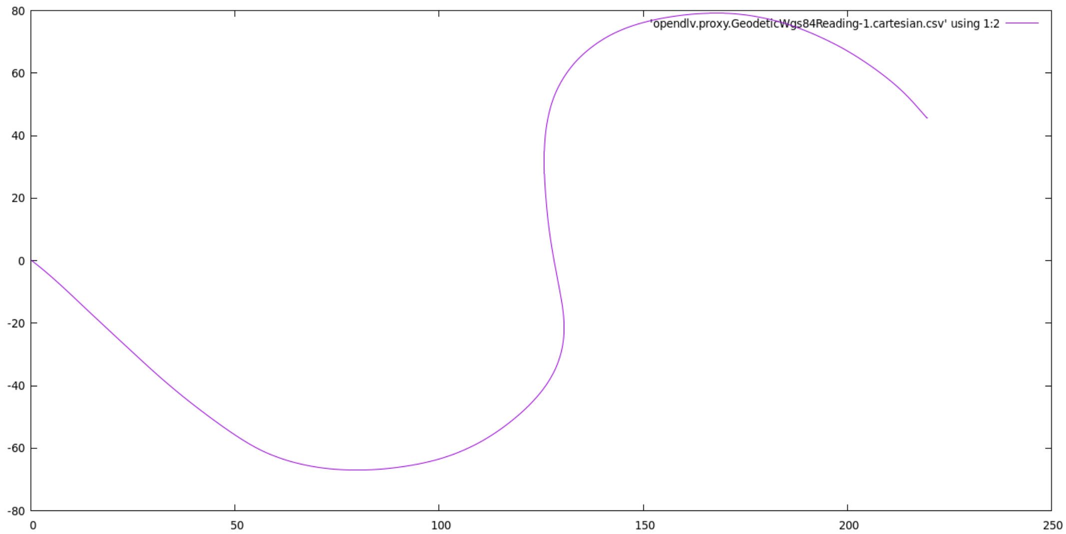


Figure 3.9: An example of GPS path of trial converted to local Cartesian coordinates.

The GA optimized the best driver models and gave out the steering angle. The GP method automatically compared the model steering output with the recorded steering signal in the data to automatically design and tune an optimal driver model. The driver model which generated the closest steering wheel angles in comparison to or matches the human driver-generated steering angle was considered to be the best-fit driver model.

4

Results

Driver 1 The estimated steering angles are plotted and compared with the recorded data in Figure 4.1 and Table 4.1. The blue line in the plots represents the human-driven steering angle, and the orange line represents the steering angle generated from the simulation result.

Driver 2 The estimated steering angles are plotted against the recorded data in Figure 4.2 and Table 4.2. The blue line in the plots represents the human-driven steering angle from the recording, and the orange line represents the steering angle from the simulation result.

Driver 3 The estimated steering angles are plotted and compared with the recorded data in Figure 4.3 and Table 4.3. The blue line in the plots represents the human-driven steering angle from the recording, and the orange line represents the simulation result. Some of the trials were not on par when compared with the rest of the two drivers. The simulation was run for all the trials of driver 3 which were completed.

GA notation Tables 4.1, 4.2 and 4.3 describe DSL sentences, fitness values and speeds for different GA optimization runs. Some notations are used:

- Type 0: Aimpoint angle
- Type 1: Aimpoint angle rate
- Type 2: Optical lever
- Type 3: Heading error
- k: Gain for respective preview point or lever offset
- d: Distance to respective point or lever offset

General observations It is clear from tables 4.1, 4.2, and 4.3, that in almost all cases, the preferred driver model is a linear combination of a Type 0 (Aimpoint angle) and Type 2 (Lever point offset) word. It is also notable that the Aimpoint angle gain is negative, and that the lever point offset gain is positive. It is also notable that no other driver model ever is chosen by the genetic algorithm.

Speed	Fitness	Type / d / k	Type / d / k
20	-6.84E-04	0 / 41.00 / -1.41	2 / 13.52 / 0.01
25	-4.48E-04	0 / 51.33 / -0.81	2 / 20.70 / 0.01
28	-4.25E-04	0 / 39.94 / -1.77	2 / 12.52 / 0.02
39	-3.62E-04	0 / 44.58 / -2.00	2 / 7.90 / 0.03
47	-4.36E-04	0 / 49.00 / -2.00	2 / 9.32 / 0.03
56	-9.38E-04	0 / 57.27 / -2.00	2 / 31.70 / 0.02

Table 4.1: DSL sentences optimized by the GA, based on the logs from Driver 1.

Speed	Fitness	Type / d / k	Type / d / k
20	-4.19E-04	0 / 41.65 / -1.38	2 / 32.46 / 0.01
26	-3.71E-04	0 / 43.28 / -1.07	2 / 20.86 / 0.01
28	-7.13E-04	0 / 39.28 / -1.55	2 / 33.98 / 0.01
39	-3.62E-04	0 / 42.24 / -2.00	2 / 35.88 / 0.01
47	-3.94E-04	0 / 44.19 / -2.00	2 / 27.00 / 0.02

Table 4.2: DSL sentences optimized by the GA, based on the logs from Driver 2.

Speed	Fitness	Type / d / k	Type / d / k	Type / d / k
21	-1.28E-03	0 / 46.24 / -1.06	2 / 27.47 / 0.02	
20	-2.56E-03	0 / 90.68 / -0.50	2 / 29.37 / 0.02	
27	-1.55E-03	0 / 87.12 / -0.79	2 / 35.49 / 0.02	
37	-7.64E-03	0 / 100.00 / -1.29	2 / 40.36 / 0.04	
44	-2.24E-03	0 / 92.81 / -0.95	0 / 1.31 / 0.08	2 / 22.84 / 0.05

Table 4.3: DSL sentences optimized by the GA, based on the logs from Driver 3.

.

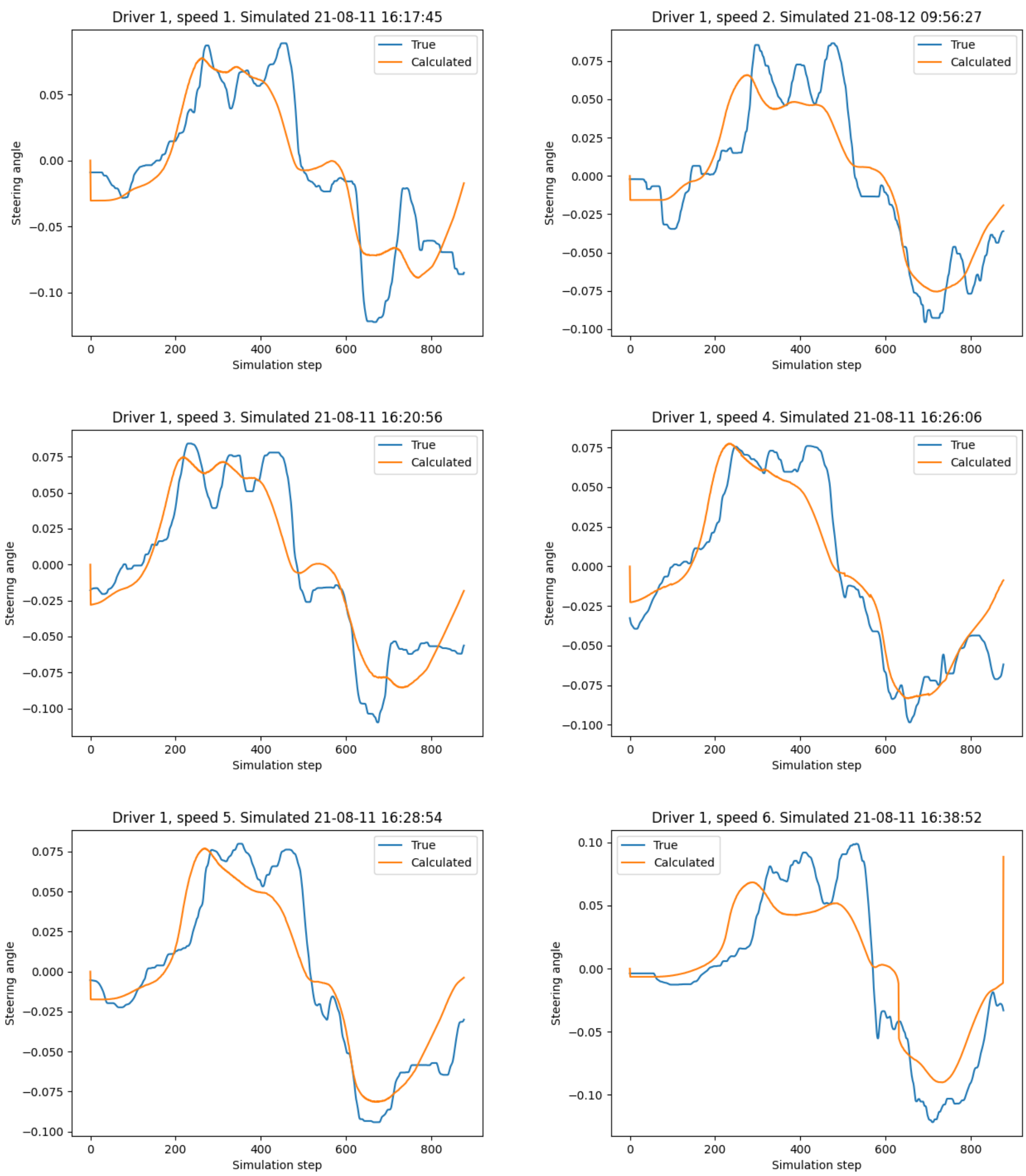


Figure 4.1: A comparison between the actual and generated steering angles for driver 1.

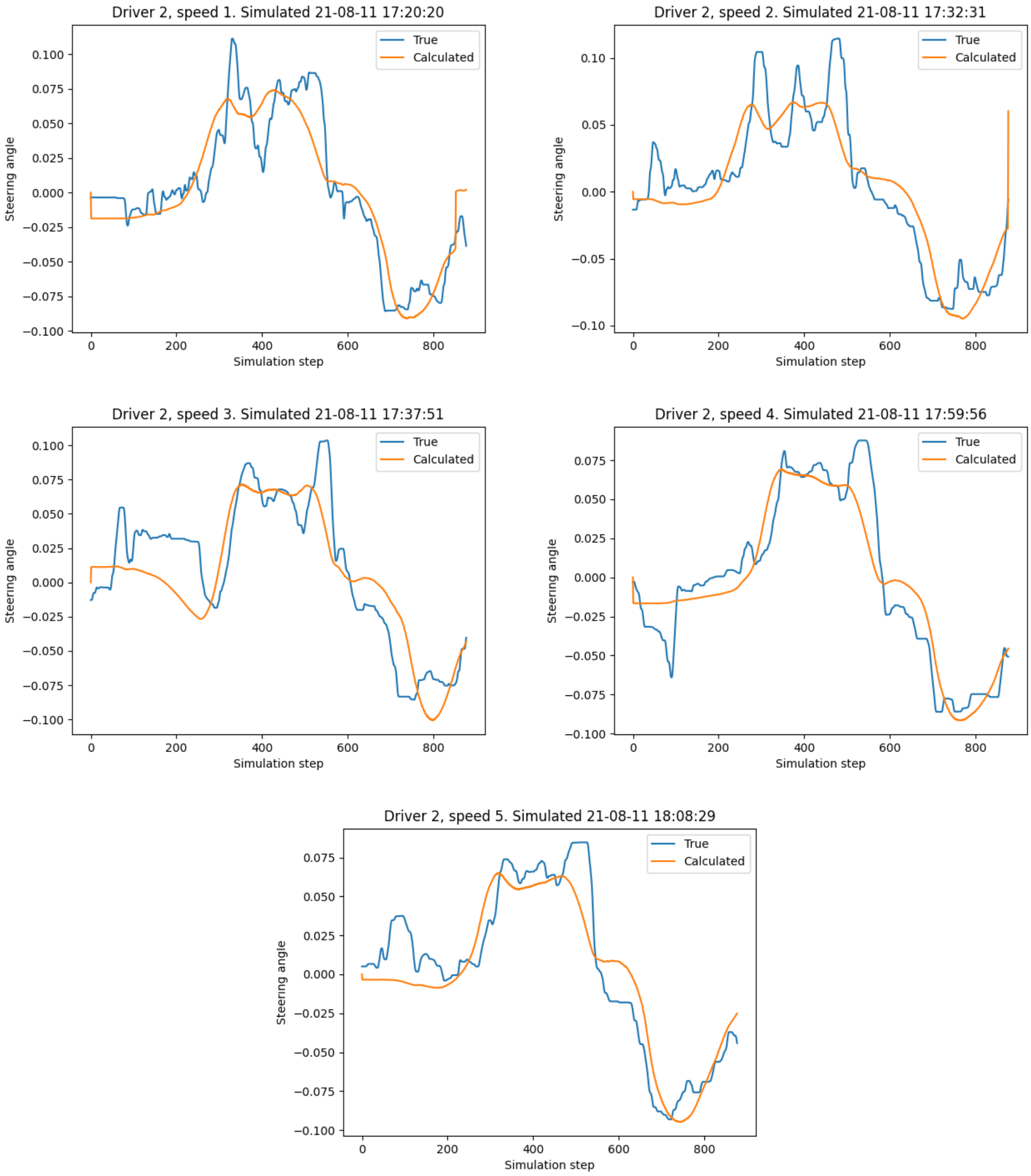


Figure 4.2: A comparison between the actual and generated steering angles for driver 2.

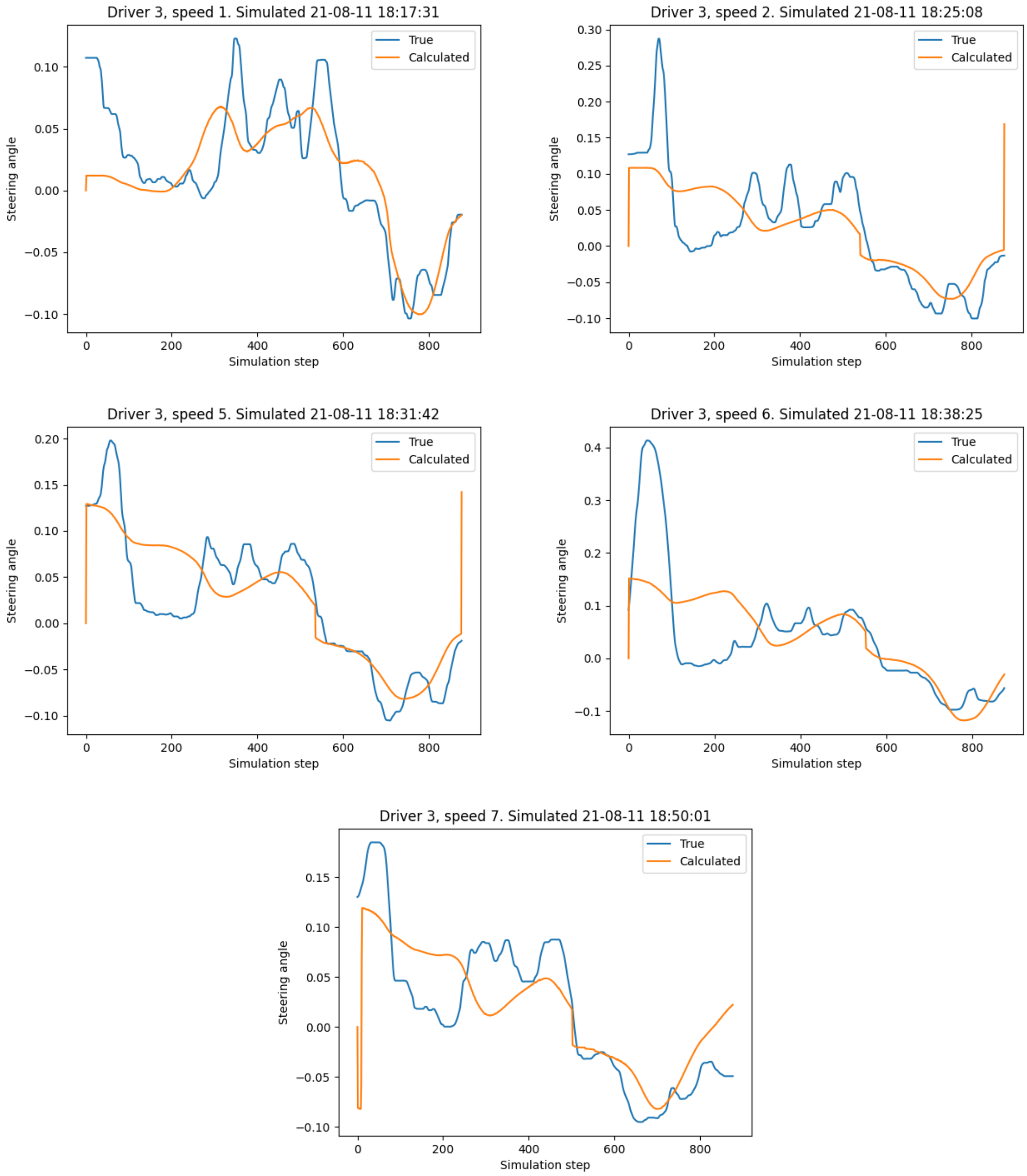


Figure 4.3: A comparison between the actual and generated steering angles for driver 3.

5
Discussion

The genetic algorithm provides DSL words with parameters that correspond to components from existing driver models. Since the fitness function used is mean square error, it can also be thought of as roughly equivalent to variance, which gives an idea of how closely the driver models model real world human driving behaviour.

5.1 Performance of genetic algorithm

The GA was run for 1000 generations, with populations of 100 individuals in each generation, an upper limit on individual length of 100. The elite size was taken to be 10, the tournament selection probability 0.5, the crossover probability 0.5, and the mutation probability 0.05.

From the resulting DSL sentences, if the DSL sentences are to be converted back to driver models, it seems like the GA explains most of the variance with a Benderius model. This is something that can be seen clearly and consistently in every single trial, as shown in the tables 4.1 4.2 and 4.3, where the word type 0 that corresponds to aimpoint angle error have by far the greatest constants in magnitude, even if adjusted for the size of the lever provided by the other DSL words.

However, the remainder cannot be ignored. The optical lever DSL words show up just as systematically as the Benderius terms, and it seems like the residual variance that is not explained by the Benderius model fits well with an optical lever.

Since the fitness function used is mean square error, which is mathematically (almost) equivalent to variance, it is possible to calculate an approximate standard deviation for the steering input error as the square root of the fitness. For example, for driver 1 trial 6:

$$f = 9.38e - 04 \Rightarrow MSE = \sqrt{f} = 0.03 \approx \sigma$$

which is approximately 1.72 degrees. This calculation will be used in the analyses below.

5.1.1 Analysis of driver 1 trial results

The steering angles generated for the first driver follow the ground truth within a range of 1-2 degrees. At greater speeds, the aimpoint distance seems to grow,

and this is quite intuitive, as it seems natural that the driver would need to know information within some set window of time ahead, which is equivalent to a linear relationship between speed and aimpoint distance. Likewise, the gain associated with the aimpoint angle error seems to be higher at higher speeds, which also makes sense as the slip angle increases with the speed of the vehicle.
Considering an example, the DSL sentence of driver 1, speed 1-
[T=0; d=40.997; k=-1.4082] [T=2; d=13.5205; k=0.00994685]

The DSL sentence has word types T=0 and T=2, which translates to aimpoint angle and optical lever respectively. The mathematical formula of the Benderius driver model largely contributes to the estimation of steering angle in this DSL sentence, but the optical lever cannot be ignored, since although the distance or length multiplied by gain is lower in magnitude, it may still serve a fine tuning purpose in addition to the Benderius term.

5.1.2 Analysis of driver 2 trial results

Similarly to driver 1, the steering angles are in the range of 1 to 2 degrees approximately, and the relationship between distance, gain and speed seem to be present here as well. The DSL sentences are overall very similar to those shown with driver 1.

5.1.3 Analysis of driver 3 trial results

Similarly to drivers 1 and 2, the steering angles are in the range of 2 degrees approximately, and the relationship between distance, gain and speed are again present. The DSL sentences are overall very similar to those shown with driver 1, with one notable exception where a second Benderius term has been added. With a combination two Benderius terms at wildly different distances, this particular DSL sentence is perhaps better understood as a Salvucci and Gray model, but since none of the other 15 runs have this term, and since the fitness values are almost an order of magnitude higher for this driver, this may also be interpreted as a failure of the GA to find a suitable model at all.

5.1.4 Observations

In the trials conducted among all the drivers, it can be observed that at the end of some of the trials, the estimated steering angles go vertically up with respect to the simulation time. This is because there is no more look-ahead left when the end of the simulation is approached, and since the driver models are based on preview points, this naturally makes this part of the dataset unreliable. It is also notable that this will affect the values of the fitness functions slightly.

The steering torque is not considered in the simulations. The steering torque is a factor dependent on the road friction, pneumatic trail, caster trail, scrub radius, and the forces acting on the tyre from the ground. The steering feel is also an added factor for the minor changes to correct or steer the vehicle. The body roll correction is also an added factor [13]. The jerks in the graphs of human driving can be explained with these factors whereas the GA estimated steering angles does not consider these factors.

The trend from all the trials observed is that the aimpoint angle error distance is higher at higher speeds and also the gain magnitude associated with it is larger at higher speeds than lower speeds.

A comment should also be made on the lengths of the generated DSL sentences. It is perhaps natural, in an experiment such at this one, to include some sort of regularization parameter to favour shorter, simpler sentences compared to longer ones. However, that was not done here. It is of course impossible, and indeed almost certainly false to claim that no DSL sentence with more than three words could achieve a lower fitness value than the ones observed in this study, but it is clear that the genetic algorithm strongly favours short sentences. Why that is, however, is for others to explore.

5.2 Overview

It should be mentioned that this report, perhaps more significantly than the results themselves, demonstrate how a genetic algorithm can be used to optimize driver models based on the DSL proposed here. This makes lateral motion control design a lot easier, when models can be added, combined, fitted and removed with less work.

The point of having the results in the GP method is to study the inner workings of common driver models, by breaking down the driver model parameters rather than just having the final result. This approach makes it possible to understand how the driver models generate the steering angles, and provides some insight into what is normally a black box. Though the argument can be made that machine learning produces better results, with lower fitness values, that has already been explored by others, and offers little insight into how driver models are ultimately constructed.

6
Conclusion

In this thesis, a detailed study of how the driver models are generated using genetic programming was done. The addition of the DSL concept to the GA for the estimation of steering angle is introduced to evaluate the existing driver models and also generate new driver models.

The project has explored the minimum set of driver model components required to generate a steering output satisfactorily close to the true steering output. These driver model components can be interpreted as components of the information a human driver uses.

The steering angle estimated is based on the cognitive perspective side only. The parameters influencing the steering angle based on the factors on the ground, the vehicle is moving on, is not considered. These factors can be added to the estimation of the steering angle for higher fidelity. The estimated steering angle based on the factors influencing the vehicle and the vehicle tyres externally, along with the genetically generated steering angle can be the best solution for the steering angle required.

In succession to this thesis, the estimation of the steering angle can be compared with the human driving recorded steering angle for a real world driving scenario. The results of this project can be used as a reference.

It is the authors' hope that the results of this project will be considered in future research of genetically generated driver models, perhaps involving the open question of why the produced DSL sentences are not longer. The genetic algorithm approach for generating driver models for on-road autonomous vehicles may also be used to generate longitudinal control models, perhaps based on additional data about the vehicle.

Bibliography

[1] Behrooz Mashadi; Mehdi Mahmoudi-Kaleybar; Pouyan Ahmadizadeh; Atta Oveisi. A path-following driver/vehicle model with optimized lateral dynamic controller, Latin American Journal of Solids and Structures On-line version ISSN 1679-7825, Lat. Am. j. solids struct. vol.11 no.4 Rio de Janeiro Aug. 2014 https://doi.org/10.1590/S1679-78252014000400004

[2] Ola Benderius, "Driver modeling: Data collection, model analysis, and optimization", 2012 THESIS FOR LICENTIATE OF ENGINEERING no 2012:11 ISSN 1652-8565, Department of Applied Mechanics, Chalmers University of Technology, Göteborg

[3] Dario D Salvucci and Rob Gray, "A Two-Point Visual Control Model of Steering," Perception, vol. 33, no. 10, pp. 1233–1248, Oct. 2004, doi: 10.1068/p5343.

[4] Michael Land and Julia Horwood 1995 "Which parts of the road guide steering?" Nature 377 339-340, Sussex Centre for Neuroscience, School of Biological Sciences, University of Sussex, Brighton BN1 9QG, UK

[5] R.S. Sharp , D. Casanova and P. Symonds (2000): "A Mathematical Model for Driver Steering Control, with Design, Tuning and Performance Results", Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility, 33:5, 289-326

[6] Ola Benderius, "Modelling driver steering and neuromuscular behaviour", Thesis for the degree of doctor of philosophy in machine and vehicle systems

[7] Stanley: The Robot that Won the DARPA Grand Challenge Sebastian Thrun, Mike Montemerlo, Hendrik Dahlkamp, David Stavens, Andrei Aron, James Diebel, Philip Fong, John Gale, Morgan Halpenny, Gabriel Hoffmann, Kenny Lau, Celia Oakley, Mark Palatucci, Vaughan Pratt, and Pascal Stang, Stanford Artificial Intelligence Laboratory Stanford University, Stanford, California 94305

[8] Markkula, G., Benderius, O., Wolff, K. et al. (2012) "A Review of Near-Collision Driver Behavior Models". Human Factors, vol. 54(6), pp. 1117-1143. http://dx.doi.org/10.1177/0018720812448474

[9] Markkula, G. ; Benderius, O. ; Wahde, M. (2014) "Comparing and validating models of driver steering behaviour in collision avoidance and vehicle stabilisation". Vehicle System Dynamics, vol. 52(12), pp. 1658-1680. http://dx.doi.org/10.1080/00423114.2014.954589

[10] Diego Jauregui and Stefan Vukman, "Automated driver modelling using genetic programming with a domain-specific language", Department of Mechanics and Maritime Sciences, Chalmers University of Technology, Göteborg

[11] Florian Bock, Christoph Sippl, Aaron Heinz, Christoph Lauer and Reinhard German ,"Advantageous Usage of Textual Domain-Specific Languages for

Scenario-Driven Development of Automated Driving Functions", Department of Computer Science Friedrich-Alexander-University, 91058 Erlangen-Nuremberg, Germany

[12] Rodrigo Queiroz, Thorsten Berger, Krzysztof Czarnecki,"GeoScenario: An Open DSL for Autonomous Driving Scenario Representation", 2019 IEEE Intelligent Vehicles Symposium (IV) Paris, France. June 9-12, 2019

[13] M. Wahde, "Biologically inspired optimization methods: An introduction, Southampton, UK; Boston, MA: WIT Press, 2008.

[14] https://www.chalmers.se/en/departments/e2/news/Pages/Revere-celebrates-five-years-of-vehicle-research.aspx

[15] Bengt Jacobson et al, "Vehicle Dynamics Compendium for Course MMF062", Vehicle Dynamics Group, Division of Vehicle and Autonomous Systems, Department of Applied Mechanics, Chalmers University of Technology, www.chalmers.se

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY