Comfort-Oriented Motion Planning Using Deep Reinforcement Learning

Master Thesis

Nishant Rajesh

Delft University of Technology

TU Delft

Comfort-Oriented Motion Planning Using Deep Reinforcement Learning

Master Thesis

by

Nishant Rajesh

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Wednesday August 31, 2022 at 9:00 AM.

Student number:	5229766	
Project duration:	February 1, 2022 – August 31, 2022	
Thesis committee:	Dr. R. Happee,	CoR-IV, TU Delft, Chair
	Dr. B. Shyrokau,	CoR-IV, TU Delft, Supervisor
	Dr. M. Alirezaei,	TU Eindhoven, Committee member
	Ir. Y. Zheng,	CoR-IV, TU Delft, Supervisor

This thesis is confidential and cannot be made public until August 31, 2024.

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

TU Delft

Cognitive Robotics

Acknowledgements

I may be the author of this thesis, but it goes without saying that this work is the culmination of the efforts of a number of people, to express my gratitude to whom I would like to take a few words.

First and foremost is my supervisor, Dr. Barys Shyrokau. He provided me with deep insights into the subject matter, while at the same time enabling me to see the big picture with his farsightedness. I could not have asked for a more knowledgeable, understanding and supportive supervisor, and I am forever grateful to him for his guidance. In the same breath, I would also like to thank Yanggu Zheng, for being my guiding light throughout this entire thesis, for his patience and wisdom, and for being a mentor and a friend.

I want to thank Sarah, for being the first person to make me feel like I have family in Delft, for giving boundless support and love to a person she had no obligation towards. I want to thank Nikhil and Pallav for their undying faith in me when I had none, for being there for me in both happiness and sadness, and for constantly inspiring me to achieve greater heights. This acknowledgement would be incomplete without thanking Atharv, Rohan, Sampada, Sreeparna and Suchdeep, without whom I cannot imagine a future now. They provided me with a new lease of life, and I cannot express how fortunate I feel to call these gems my friends.

And finally, I want to express my deepest, most sincere and heartfelt thanks to my family. To Maithili, for the joy and hope she brings. To my brother Nisheeth, for being the rock-solid foundation upon which I have built my life, and for being everything that a person could ever wish for in a brother, and much more. And to my parents, for blessing me with the privilege of being their son. They have given me love beyond what I could repay in multiple lifetimes, but as a small offering, I would like to dedicate this thesis to them.

Nishant Rajesh
Schiedam, August 2022

Abstract

Motion sickness is a common phenomenon, with close to two-thirds of the population experiencing it in their lifetime. With the advent of automated vehicles in the market, it is anticipated to become an even greater problem as the passengers face a lack of predictability of motion and loss of control over the vehicle. This could nullify the host of possible benefits that automated vehicles propose to offer, and therefore affect their acceptance among potential users.

It is well known that the nauseogenicity of imposed motion is dependent on the frequency content of endured accelerations, with low-frequency accelerations being the primary contributor. This thesis presents a motion planning algorithm targeted towards minimization of motion sickness among passengers of automated vehicles, through targeted reduction of low-frequency accelerations. A Deep Reinforcement Learning (DRL) framework was utilised along with the design of a custom environment and a reward function which incorporates a measure of nauseogenicity of the planned trajectories. The frequency shaping effect of the reward function was evaluated by comparing against a DRL agent trained to optimize general motion comfort described by total acceleration energy. It was found that the nauseogenicity was reduced by 9.6% with the proposed DRL agent.

Further, on-road trials were performed with human drivers to establish a benchmark of driving comfort. The performance of the DRL agent was compared to human drivers as well as against an optimization-based motion planner that computationally maximizes the reward function. The DRL planner displayed comparable performance to the human drivers, and was within 10 to 15% of the discomfort levels of the optimization-based planner for a range of travel times. Meanwhile, the DRL planner offered notable improvements in computational efficiency, taking 1-2 ms to generate a sub-optimal trajectory, as opposed to approximately 5 s as taken by the optimization-based planner.

Contents

Acknowledgements i

Abstract ii

1 Introduction 1

2 Journal Paper 4

3 Results and Discussions 17
3.1 Additional Results . . . 17
3.2 Discussion . . . 17
3.2.1 Environment Design . . . 17
3.2.2 Evaluation of Real-World Benefits . . . 21
3.2.3 DRL Issues . . . 21

4 Conclusions and Future Work 23
4.1 Conclusions . . . 23
4.2 Future Work . . . 23

A Deep Reinforcement Learning 24
A.1 Basic Concepts . . . 24
A.2 Value Approximation Based Algorithms . . . 25
A.3 Policy Based Algorithms . . . 26
A.4 Proximal Policy Optimization . . . 27

B Custom Environment Definition 28
B.1 Training Environment . . . 28
B.1.1 Road Profile Generation . . . 28
B.1.2 Motion Profile Calculation . . . 29
B.2 Evaluation Environment . . . 30

C Discomfort Evaluation 33
C.1 Motion Sickness Dose Value . . . 33
C.2 Frequency Weighting Filters . . . 33
C.3 Filter Implementation . . . 35

References 37

List of Figures

3.1 Frequency spectrum comparison of longitudinal accelerations about roundabout 1 . . . 18
3.2 Frequency spectrum comparison of lateral acceleration about roundabout 1 18
3.3 Frequency spectrum comparison of longitudinal accelerations about roundabout 2 . . . 19
3.4 Frequency spectrum comparison of lateral accelerations about roundabout 2 19
3.5 Mean episode rewards vs training time for DRL agent with action space of size 8. . . . 20
3.6 Mean episode rewards vs training time for DRL agent with action space of size 14. . . 21

B.1 Satellite image of roundabout 1 31
B.2 Satellite image of roundabout 2 31

C.1 Motion sickness frequency weightings for lateral acceleration 34
C.2 Band Pass filters used for calculating discomfort term 35
C.3 Step response of longitudinal frequency weighting filter 36
C.4 Step response of lateral frequency weighting filter 36

1

Introduction

Motion sickness is a nausea syndrome observed in humans when they are subjected to passive motion stimuli. It manifests in the form of symptoms such as drowsiness, dizziness, sweating, headaches, stomach awareness, nausea and so on, and can be induced by actual or illusory motion [1]. Motion sickness has been affecting humans ever since the existence of means of transportation, with the earliest recorded studies dating back to 800 BC, as experienced by Greek sailors [2]. Mobility has become much more commonplace and accessible in the present, and consequently, so has the occurrence of motion sickness in the general population. In their lifetime, nearly two out of three people experience motion sickness [1]. An international survey had comparable findings for carsickness in particular, with 46% of the participants having suffered from some degree of carsickness in the past five years, with the number increasing to 59% on including childhood experiences [3]. Children between the ages of 10-12 years are the most susceptible to car sickness [4].

With the advent of automated vehicles, it has been hypothesised that there may be a shift from car sickness, to autonomous car sickness [5]. There has been significant progress in the development of automated vehicles over the past few years, and they remain a field of promising research. They promise to revolutionise the mobility industry, by eliminating accidents and mishaps caused by human errors. Along with the improved safety, they also are envisioned to improve traffic efficiency and vehicle fuel efficiency. However, from a user perspective, one of the most attractive benefits is the prospect of being freed from the driving task, allowing them to engage in a multitude of non-driving tasks. The time freed up from driving would let users to take up a plethora of tasks such as texting messages, eating and drinking, surfing the Internet, performing office tasks and so on [6]. Automated vehicles would also make road travel much more accessible to children and the elderly, the parts of the population which are unable to drive.

The merits of automated driving, while significant, could be curtailed by the increased prevalence of car sickness in passengers. As the driver transitions from an active role to a passive one, they become much more susceptible to suffering from car sickness due to a lack of perceived controllability, attention diversion and predictability [7]. This would be further aggravated by engaging in non-driving tasks, as the passengers would have insufficient visual information and also lack an Earth fixed horizon, both of which are known to increase motion sickness symptoms [8]. The increased accessibility of automated vehicles to children also requires taking into consideration carsickness effects, since they are particularly susceptible. It is evident from these facts that carsickness is indeed a matter of concern in automated vehicles, as it could easily nullify many of the proposed user benefits, and consequently affect the acceptance of automated vehicles among the general population. It is therefore critical to incorporate mechanisms within automated vehicles which prevent or mitigate the effects of motion sickness in passengers.

The underlying cause of motion sickness is still not completely understood, however, multiple theories have been proposed which attempt to explain the mechanisms causing motion sickness in humans. The most prevalent theory in literature is the sensory conflict theory [1, 9]. It proposes two premises

for the development of motion sickness. The first is that there should be a conflict between the motion signals as observed by the visual system, the vestibular system and the non-vestibular proprioceptors, and the sensory conflict should be at variance from previously experienced motion stimuli. It is not sufficient for there to be a sensory conflict, as the nervous system undergoes a sensory rearrangement and continued exposure to the conflict signal leads to habituation. The second premise is that the vestibular system needs to be involved in the sensory conflict, directly or indirectly, for the causation of motion sickness. This implies that the motion stimuli need to include angular or linear accelerations, as constant velocities cannot be sensed by the vestibular system.

O'Hanlon and McCauley investigated the occurrence of nausea in subjects as a function of imposed accelerations and their frequencies [10]. They found that for oscillations in the vertical direction, the primary cause of motion sickness were low frequency accelerations in the range 0 to 0.5 Hz. Accelerations around a frequency of 0.16 Hz were found to be the most nauseogenic. It was also seen that the incidence of sickness increased monotonically with the magnitude of imposed accelerations. Similar response was found for lateral and longitudinal accelerations with slight variations in the frequency response [11, 12]. A study on passengers of public buses verified that the causation of motion sickness was primarily due to low frequency accelerations [4]. In particular, low frequency lateral accelerations had the highest correlation with the incidence of nausea in passengers, followed by longitudinal accelerations. There was no significant correlation with vertical accelerations, as well as roll, pitch, and yaw motions. The study also concluded that to minimise car sickness among passengers, the driving style itself needed to be altered, and merely changes in vehicle design to alter dynamic characteristics of the vehicle would not suffice in reducing motion sickness.

With regards to automated vehicles, the driving style of the vehicle is defined by the motion planning layer. While there has been significant research into motion planning for improvement of general passenger comfort in automated vehicles [13–16], the research into motion sickness mitigation is relatively nascent. Most existing literature tries to deal with motion sickness in the vehicle control layer [17–19]. Vehicle control is generally reactive in nature, and even predictive methods typically work by attempting to track a reference motion plan defined by a planning algorithm over a short horizon. This approach is not ideal for the objective of minimising low frequency accelerations in particular, since they are characteristic of the predefined trajectory. Therefore, they need to be dealt with in the motion planning layer itself. The attempts in literature which do try to specifically target motion sickness in the motion planning layer typically make use of optimization-based techniques [20–22]. While attractive for their relative ease of implementation and guaranteed optimality, these methods are computationally demanding, and real-time implementation on on-board vehicle computers may be problematic.

Deep Reinforcement Learning (DRL) offers an attractive alternate possibility to optimization-based methods, as it requires minimal online computational effort. In DRL, a Neural Network (NN) is trained to perform a task in an environment (real or virtual), by providing positive or negative reinforcement through rewards or penalties respectively. while learning to perform the task itself, the agent requires long training times and consequently, significant computational resources. However once trained, the NN requires minimal computational power to implement online. DRL does not suffer from the drawbacks of typical data-driven approaches, where large amounts of high quality data are required to be able train neural networks to perform the desired task. Data collection is especially difficult for our problem, as finding 'good drivers' which minimize motion sickness for all passengers is not trivial. DRL circumvents this issue by collecting training data through interactions with a simulation environment, and therefore only the design of a representative environment is required.

DRL for automated vehicles has been well researched [23–27], with application to motion planning as well [27]. DRL has also been shown to effectively deal with long term dependencies when combined with Monte Carlo Tree Search algorithms, or by using Long Short Term Memory Neural Networks for the state representation [28–30], and is therefore particularly attractive for motion sickness mitigation where low frequency accelerations are the most significant.

In this thesis, a Deep Reinforcement Learning motion planner has been designed, with the objective of minimising motion sickness in passengers of automated vehicles. A custom environment is designed, representative of the real road profiles the agent is expected to encounter. The objective of carsickness mitigation through the reduction of low frequency accelerations is incorporated into the reward function of the DRL agent. To investigate the ability of the DRL agent to learn to mitigate the undesirable

frequencies, the frequency response of the agent is compared to another DRL agent which is trained to improve general passenger comfort without regard to the frequency of accelerations.

To further ascertain how the DRL motion scheme compares to a human driver, a human baseline performance is established through on-road experiments with human drivers. The performance of the DRL agent is evaluated in a high fidelity IPG CarMaker environment, and benchmarked against the human drivers. The nauseogenicity of the respective motion plans have been investigated over a range of travel times, to emulate different driving styles encountered with human driver.

The thesis is structured as follows. Chapter 2 consists of the journal paper summarising the work of the thesis. Chapter 3 elaborates further on the results mentioned in the journal paper, and discusses some limitations of the work. Chapter 4 concludes the findings of the thesis and discusses the scope for future work. The appendices lay out the methods used in further detail.

2

Journal Paper

Comfort-Oriented Motion Planning Using Deep Reinforcement Learning

***Abstract*—Automated vehicles promise numerous advantages to users in the form of improved safety, efficiency, and productivity. The proposed benefits of automated vehicles can however be overshadowed by the increased susceptibility of passengers to motion sickness because of them taking on increasingly passive roles and simultaneously engaging in non-driving tasks. Increasing attention is being paid towards the design of motion regimes for automated vehicles which mitigate carsickness, while maintaining reasonable travel times. In this work, a Deep Reinforcement Learning (DRL) approach has been used to plan vehicle trajectories, with a focus on minimizing low-frequency accelerations which are known to be the primary cause of motion sickness. This is achieved through incorporating a frequency weighted discomfort term into the reward function of the training environment.**

The ability of the DRL agent to target undesirable frequencies in the planned accelerations is investigated by comparing with a DRL agent trained to improve general comfort, and with an optimization-based motion planner. A reduction of discomfort by 9.6% is achieved as compared to the benchmark DRL agent. The motion planning method is further validated by comparing the accelerations generated by the motion plans, with trajectories generated by human drivers, on two actual roundabout scenarios. The results demonstrate that the DRL motion planner achieves comparable performance to human drivers, while offering massive improvements in online computation time compared to optimization-based planners.

***Index Terms*—Motion Planning, Motion Sickness, Deep Reinforcement Learning, Automated Driving, Proximal Policy Optimization**

I. Introduction

AUTOMATED vehicles are a field of intensive research currently in the automotive domain, with significant strides in their development in the recent years. The increased attention towards the development of automated driving is owing to the potential benefits they offer in terms of improved safety, higher traffic efficiency and increase in user productivity. Being freed from the responsibility of driving the vehicle, users of automated vehicles are expected to engage in numerous non-driving tasks ranging from conversing with co-passengers and listening to music, to texting, eating, drinking, websurfing and so on [1]. In order for the passengers to be able to perform such tasks, it is imperative that automated vehicles provide a high level of driving comfort.

It is anticipated that with the advent of complete automation in cars, there would be an increased susceptibility of passengers to carsickness [2]. This could be due to a multitude of reasons. The driver would take on a much more passive role, especially with higher levels of automation, which is well known to increase motion sickness [3], [4]. Many automotive companies are also re-imagining vehicle cockpit design, unveiling concepts with office like environments, rearward facing seats and passengers facing each other. Combined with the passengers engaging in non-driving tasks, this would lead to a lack of a stable visual horizon, and lower predictability of the direction of motion. The combined effect of all these factors may very well lead to significant increase in the occurrence of carsickness, posing a substantial threat to the envisioned benefits of automation, and consequently to the acceptance of automated vehicles among customers.

Motion sickness is a nausea or vomiting syndrome in healthy subjects arising from illusory or actual passive self motion. There can be numerous and varied symptoms of motion sickness ranging from drowsiness and fatigue to stomach awareness and nausea [5]. The most widely accepted theory explaining motion sickness is the sensory mismatch theory, which postulates that motion sickness arises from the conflict between anticipated and sensed motion stimuli [3]. The Central Nervous System (CNS) maintains an internal model of the dynamics of the human body, which estimates the spatial orientation of the body by fusing information from motor outflow and noisy sensory signals. The conflict between these efference signals with the polysensory afference signals obtained from the sensory organs is used to update and improve the internal observer model, but also gives rise to motion sickness.

The incidence of motion sickness is predominantly caused by low frequency accelerations (<0.5 Hz), with the effect peaking around a frequency of 0.2 Hz for vertical accelerations [6], [7], with a similar peak for longitudinal oscillations [8]. For lateral accelerations, it was found that the incidence of motion sickness was independent of frequency from 0.0315 to 0.2 5Hz, followed by decreasing intensity with higher frequency levels [9]. The frequency weighting filters to predict incidence of nausea for lateral and vertical oscillations have been shown in Figure 1. As is evident from the frequency weighting, in order to efficiently inhibit the incidence of motion sickness, it is necessary to deal with the low frequency accelerations. This would in turn require motion planning over longer time horizons to accurately predict the low frequency acceleration components.

While there has been some research in the design of automated vehicles to mitigate motion sickness among passengers such as through the layout of the seating arrangement [10] and through provision of audio and visual cues [11], [12], they do not address the underlying cause of carsickness, which is the motion regime itself. Koppa and Hayes found the magnitudes of accelerations generated by different drivers to be independent of the vehicle characteristics, and were heavily influenced by the driving style of the individual driver [13]. Further strengthening the link between driving style and motion sickness, Turner and Griffin found that the driver was heavily implicated in the generation of motion sickness

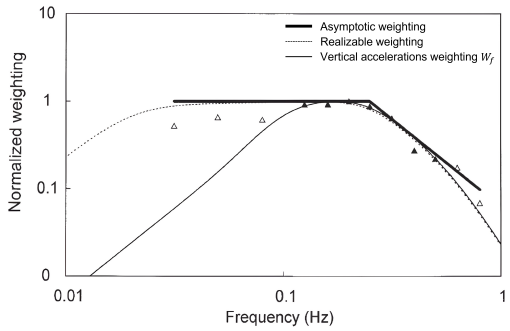

Fig. 1. Asymptotic and realizable frequency weightings for lateral acceleration [9]. The triangles are the measured instances on which the weighting filters have been fit

among passengers, with low frequency lateral accelerations being primarily responsible for nauseogenic symptoms [14]. No significant correlation between vertical, roll or pitch motion and motion sickness was established. Based on these findings, it can be said that to effectively combat the occurrence of motion sickness in automated vehicles, the vehicle motion itself needs to be planned with particular attention to lateral and fore-aft accelerations.

Passenger comfort as an objective in motion planning has been studied extensively through methods such as planning of smooth paths using clothoids, bezier curves and polynomial splines, or through the optimization of acceleration and jerk values [15]–[20]. However, there is very limited research with regards to directly addressing motion sickness through motion planning. Htike et al. formulated the motion planning problem as an optimal control problem, with the objective of minimising the Motion Sickness Dose Value (MSDV), and investigated the relation between sickness levels and travel time [21]. Li and Hu also attempted to address the motion sickness minimization problem in motion planning through the formulation of an optimization problem, but used a frequency shaping approach. However, they only utilized a high pass filter, relying on the vehicle actuator limits to filter out higher frequencies [22]. Ukita et al optimized the vehicle trajectory as well as the gains of the path following controller for a lane change maneuver, using vertical conflict values predicted by the 6 Degree of Freedom Subjective Vertical Conflict (SVC) model [23]. All these works on motion sickness minimization through means of motion planning focus on solving optimization problems. Classical optimization techniques however, are demanding on computational resources, and given the limited on-board computational power available on automated vehicles, it could be challenging to solve the motion planning problem in real-time. In particular, including the frequency weighting filters into the planning further introduces non-linearities into the problem, making it more difficult to solve.

Machine learning techniques could be an attractive alternative approach to the problem, as they can effectively shift the bulk of the computational demand offline, and require negligible on-board resources. They have successfully been applied to a plethora of engineering problems ranging from object detection and image classification to speech recognition and recommendation systems. However, the more widely used supervised learning techniques cannot be easily applied to our purpose of motion sickness minimisation, as they would require massive amounts of labelled training data. Training data is difficult to collect for motion sickness since it would require large scale data collection from human subjects imposed to sickening driving regimes. In addition, the inherent variability among humans of susceptibility to motion sickness could pose problems in deciding the nauseogenicity of imposed motion.

Deep Reinforcement Learning (DRL) is a machine learning paradigm which combines the fields of Deep Learning and Reinforcement learning. It can prove to be a viable alternative to optimization-based methods in motion planning, as it does not require any labelled training data, and only needs a representative training environment. The motion sickness models from literature can be incorporated into the environment to define the reward function. DRL has already been applied effectively to various levels of motion planning problems, with the notable advantage of requiring relatively low computational requirements for the trained network [24]. Some automotive applications of DRL include behavioural decision making, path planning to end-to-end vehicle control [25]–[29]. DRL has also been shown to be able to successfully capture long-term dependencies in systems when applied in conjunction with methods such as Monte Carlo Tree Search and Long Short Term Memory Neural Networks [30]–[32]. This could be interesting to explore with regards to capturing the effects of low frequency accelerations in carsickness.

The contribution of this paper is a DRL approach to motion planning that minimizes motion sickness among passengers of automated vehicles by optimizing the vehicle trajectory to reduce nauseogenic accelerations. In particular, the ability of DRL to target and minimize low frequency longitudinal and lateral vehicle accelerations is studied. This was done by comparing the frequency domain performance of a DRL agent trained to minimise frequency weighted accelerations to an agent trained on unweighted accelerations. The performance of these agents is also compared to a planner solving the respective environments using an optimization-based technique, to measure how close the agents come to 'solving' the designed training environment.

To further establish the applicability of DRL to motion planning with regards to motion sickness, experimental trials with human drivers were carried out on an actual road section to establish a baseline of human performance. The nauseogenicity of the trajectories followed by the human drivers is compared to those generated by DRL agents, over a range of travel times to account for varying driving styles among humans.

The paper is structured as follows. Section II gives a basic overview of DRL and details the setup for training and evaluating the agent. In section III, the experimental setup for establishing the human baseline has been explained. Section IV details the results of the simulation and the comparisons between the human drivers and the trained agent, with the final conclusions of the study in section V.

II. DRL Training Environment

In this paper, the Proximal Policy Optimization (PPO) algorithm was used to train a DRL agent to plan the motion profile for randomly generated road sections. The complete road information was made available to the agent as the state of the environment. The agent was rewarded based on a weighted sum of a discomfort measure and travel time. This study focuses on the applicability of DRL to motion sickness mitigation, and therefore it was assumed that there is no interaction with other road users. This section provides a brief overview of DRL, followed by a description of the environment, the state and action space, the reward function, and the training algorithm used.

A. Deep Reinforcement Learning

Deep Reinforcement Learning is a sub-field of machine learning algorithms, in which an agent learns to navigate a task through a process of trial and error. Reinforcement Learning involves an agent operating within an environment which takes actions leading to eventual rewards, with the objective of the agent being the maximisation of long term rewards. In Deep Reinforcement Learning, this agent is approximated by a Deep Neural Network, and hence the name.

At time-step k, the environment is modeled as a system with a state transition function

$$s_{k+1} \sim P(.|s_k, a_k) \tag{1}$$

where $s_k \in S$ is the state, and $a_k \in A$ is the action taken by the agent. The agent acts according to a policy π_θ parameterised by $\theta \in \mathbb{R}^K$, which is given as

$$a_k = \pi_\theta(a_k|s_k) \tag{2}$$

A series of actions taken by the agent following the policy π till a terminal state is reached, is called a rollout or a trajectory $\tau = [s_{0:H}, a_{0:H}]$. H is the horizon, and the steps from initiation s_0 to the terminal state s_H form an episode.

For every action the agent takes, the environment returns a scalar reward r, which is modeled by a reward function

$$r_k = R(s_k, a_k) \tag{3}$$

The expected value of the accumulated reward over a period of time is called the return $J(\theta)$

$$J(\theta) = \mathbb{E}\{\sum_{k=0}^{\infty} \gamma^k r_k\} \tag{4}$$

where $\gamma \in [0, 1)$ is the discount factor. The agent interacts with the environment and samples trajectories, with the objective of learning an optimal policy π_θ^* which maximises the expected return.

Most of the common DRL algorithms are based on some form of policy gradient, and parameter update using gradient ascent

$$\theta_{h+1} = \theta_h + \alpha_h \Delta_\theta J(\theta = \theta_h) \tag{5}$$

where h is the update step of the policy. α_h is the learning rate for updating the weights of the network. The algorithms which use a gradient estimate to perform the parameter update are known as REINFORCE algorithms [33]. The REINFORCE algorithms while being simple to implement, suffer from instability during training, low sample efficiency and a lack of robustness [34].

The performance of the REINFORCE algorithm with regards to sample efficiency and reliability has been shown to be significantly improved through the use of a clipped objective function [34]. The clipped objective estimates a pessimistic lower bound on the value of the policy performance, which ensures that the gradient update steps do not become large enough to lead to worse performing policies. This algorithm is known as Proximal Policy Optimization (PPO). The PPO algorithm can deal with continuous state and action spaces, offers ease of implementation, and reliable performance, and therefore was used for our trajectory planning problem.

B. Environment and Observation Space

The agent was trained in a custom OpenAI Gym environment. In order to ensure that the agent learns to plan comfortable paths for a wide range of scenarios, random road profiles were generated for training. In each episode, the total length of the road profile L was kept fixed, with intermediate sectors of constant curvature. The road profile was constructed with straight and circular sections, maintaining a continuous first derivative.

The initial and final sectors were straight paths, with each of the remaining sectors having curvatures κ_i sampled randomly from the uniform distribution $\mathcal{U}_{[\kappa_{min}, \kappa_{max}]}$. The length of each sector was obtained by partitioning the total length of the road into sectors, again in a manner to ensure that the lengths form a uniform distribution $\mathcal{U}_{[l_{min}, l_{max}]}$. The vehicle velocity was initiated with a random velocity also sampled from a uniform distribution $\mathcal{U}_{[v_{min}, v_{max}]}$.

The road was assumed to be a constant width throughout the entire section. The environment was assumed to be completely observable, and the state vector was defined as follows

$$s = [\kappa_{0:n-1}, l_{0:n-1}, y_0, v_0] \tag{6}$$

where $\kappa_{0:n-1}$ is the array of curvatures of the road sectors, $l_{0:n-1}$ are the lengths of the respective sectors, and y_0 and v_0 are the initial lateral position and longitudinal velocity of the vehicle respectively. Since the curvatures and lengths of the road profile, and the vehicle velocity can take any value within the defined limits, the state space is continuous in nature. The state contains the complete information of the vehicle and the road required for the agent to plan the vehicle trajectory. The states and observations space were normalised to lie between $[-1, 1]$, which ensured that the different quantities were scaled appropriately, and the weights in the neural network were not skewed due to different orders of magnitude of the state variables.

C. Motion Definition and Action Space

The vehicle motion was defined in terms of its position and velocity, and it was assumed that a path following controller would be used to follow the defined trajectory. The position of the vehicle was defined as the lateral deviation with respect to

the centreline of the road, measured radially along the centre of curvature of the respective road sector. The longitudinal velocity at each point was assumed to be tangential to the road profile.

It is imperative to ensure continuous and smooth trajectories, in order to have smooth acceleration profiles. To ensure that the planned trajectories are smooth, a cubic spline implementation was utilised to approximate the velocity and position profiles. The reference position was described as a cubic function of the distance travelled along the centerline of the path. The reference position is calculated as the lateral deviation y from the centerline, given by the following equation

$$y_i(u) = a_{y,i}u^3 + b_{y,i}u^2 + c_{y,i}u + d_{y,i} \quad i = 0, \ldots, k-1 \quad (7)$$

where $u \in [0, 1]$ is a normalised distance parameter, 0 and 1 at the beginning and end of each sub-interval P_i of the spline respectively. $a_{y,i}$, $b_{y,i}$, $c_{y,i}$ and $d_{y,i}$ are the cubic spline coefficients for the i^{th} polynomial P_i. k is the total number of cubic polynomials which compose the spline. The coefficients were calculated to satisfy the following boundary conditions

- The first derivative at beginning and end of each polynomial is continuous

$$P_{i-1}^{(1)}(1) = P_i^{(1)}(0) \quad i = 1, \ldots, k-1 \quad (8)$$

- The second derivative at beginning and end of each polynomial is continuous

$$P_{i-1}^{(2)}(1) = P_i^{(2)}(0) \quad i = 1, \ldots, k-1 \quad (9)$$

- At the start and end of the road, the first derivative is zero. This ensures an initial and final heading along the road direction, and zero initial and final longitudinal acceleration

$$P_0^{(1)}(0) = P_{k-1}^{(1)}(1) = 0 \quad (10)$$

The road profile is distributed into k control points or knots, the positions along the length of the path where the agent predicts the lateral position of the vehicle. The velocity profile is also calculated in a similar fashion, by knots placed at the same positions as the spline used to calculate lateral positioning.

$$v_i(u) = a_{v,i}u^3 + b_{v,i}u^2 + c_{v,i}u + d_{v,i} \quad i = 1, \ldots, k-1 \quad (11)$$

Together, the predicted values at the control points for both position and velocity comprise the action space of the DRL agent. The action space of the agent is therefore given as follows

$$a = [y_1(0), y_2(0) \ldots, y_{k-1}(0), y_{k-1}(1), \\ v_1(0), v_2(0) \ldots, v_{k-1}(0), v_{k-1}(1)] \quad (12)$$

Similar to the state and observation space, the action space was also normalised to lie within the bounds $[-1, 1]$. The bounds for normalization were decided based on the minimum and maximum speed limits, and the limits on the lateral deviation from centerline were enforced to prevent the vehicle exiting the lane. Considering a road width of 3.3 m, and a typical vehicle width of 2.1 m, the control knots of the spline were limited to a maximum deviation of 0.5 m from the centerline to ensure that the vehicle does not exceed the road boundaries. Inside built up areas in the Netherlands, the speed limit is 50 km/h, which is the constraint we used. The lower speed limit was kept at 18 km/h.

The knot vectors along the length of the road were placed so as to obtain equal partitions of the total length of the road. A point to be noted is that a minimum length of the road sectors (each sector with constant curvature) was enforced in order to ensure that each sector contained at least one spline knot vector. This constraint ensured that for every corner the spline could accommodate a change in direction corresponding to change in curvature of path.

D. Reward Function

In order to enable the agent to learn to plan paths which minimize motion sickness, while also optimizing travel time, it was imperative to design a reward function which incorporates these objectives appropriately. To take into consideration the vehicle accelerations, a discomfort term D was defined as the integral of the squared accelerations undergone by the vehicle, over the entire duration of the motion.

$$D = \int_0^T (a_x^2 + a_y^2)dt \quad (13)$$

Where a_x and a_y are the longitudinal and lateral accelerations of the vehicle respectively. T is the total travel time required for the vehicle to traverse the planned trajectory.

Since our goal is to selectively minimize accelerations with the most significant contribution to the generation of motion sickness in passengers, the accelerations were weighted using a frequency weighting filter prior to calculating the discomfort term. As can be seen from the Figure 1, for lateral oscillations, accelerations in the frequency range 0.02 Hz to 0.25 Hz have the most significant contribution towards inducing motion sickness in passengers, with the weighting independent of frequency of excitation [9]. Incidence of nausea has been shown to peak around 0.2 Hz for longitudinal oscillations, with the frequency dependence dropping off with higher and lower frequencies [8].

To incorporate these findings into the discomfort term, two band pass filters were constructed for lateral and longitudinal accelerations respectively. The cut-off frequencies for the lateral frequency filter were defined at 0.02 Hz and 0.25 Hz, and at 0.15 Hz and 0.25 Hz for the longitudinal frequency filter. The band pass filter is constructed as follows

$$BP(s) = \frac{1}{\tau_1 s + 1}\frac{s}{\tau_2 s + 1} \quad (14)$$

where τ_1 and τ_2 are the time constants of the low and high pass filters respectively. To ensure that neither of the lateral or longitudinal accelerations are weighted preferentially, the peak gain of the filters was adjusted to attain equal area under the curve for the frequency range 0 to 1 Hz. The band pass filters have been shown in Figure 2.

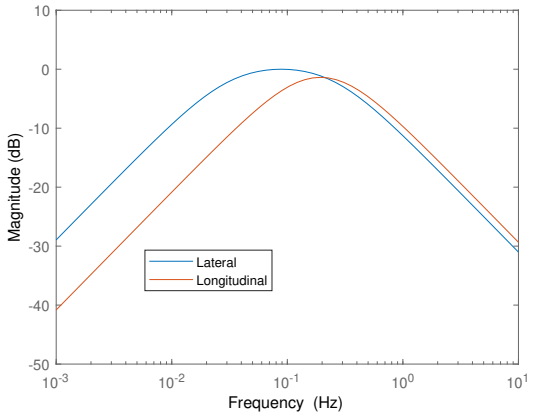

Fig. 2. The band pass filters used for weighting lateral and longitudinal accelerations to calculate the discomfort term

The filters were converted to state space models and then discretised in order to be applied to the calculated accelerations a_x and a_y. A 30 s cooldown period with zero accelerations was implemented to take into account the long tail effect of the low pass filter. The output accelerations from the filter were continued to be penalized during the cooldown period.

The overall reward was taken as a weighted sum of the discomfort D and the travel time T, to prevent the agent from learning to slow the vehicle excessively to reduce acceleration values.

$$R = W.T + D \tag{15}$$

where W is a weighing factor. The higher the value of W, the more the importance given to travel time, which would encourage the agent to plan faster trajectories at the cost of higher discomfort D. To calculate the accelerations and travel time, the path was discretised into stations spaced 1m apart along the length of the road. At each station k, the velocity v_k, and the waypoints on the path to be followed, y_k were determined using their respective spline functions as derived in the previous subsection. Using the coordinates of the stations, the respective coordinates of the waypoints were found, which was then used to calculate distance d_k between subsequent waypoints. The curvature of the path κ_k was obtained by differentiation. The point mass model was then utilized to determine the remaining values to obtain the value of the reward

$$\Delta T_k = 2d_k/(v_{k+1} + v_k) \tag{16}$$
$$a_{x,k} = (v_{k+1}^2 - v_k^2)/2d_k \tag{17}$$
$$a_{y,k} = \kappa_k(v_k + a_{x,k}\Delta T_k)^2 \tag{18}$$
$$\Delta D_k = (a_{x,k}^2 + a_{y,k}^2)\Delta T_k \tag{19}$$
$$R = \sum_{k=1}^{N-1}(W\Delta T_k + \Delta D_k) \tag{20}$$

The use of the point mass model ensures that the computational requirements during training are at a minimum as compared to more complex vehicle models. It also leads to a more general trajectory planner which can be implemented on a range of vehicles, as it does not depend on vehicle parameters.

In addition to the reward based on accelerations, to constrain planned accelerations within the traction capabilities of the vehicle, a penalty of -1000 was imposed on the agent if the total vehicle accelerations exceeded a value of $1g$ at any timestep.

E. DRL Agent Training

The training consisted of single step episodes, with the agent receiving initial the initial state information s_0, as defined in equation 6, from the environment, predicting knot vectors for the entire path in a single step, and receiving the corresponding reward. As described in section II-A, the PPO algorithm was used to train the agent. The standard PPO implementation from the stable baselines3 library [35] was used. The hyperparameters have been listed in Table I. The hyperparameters listed were optimized by a search using Optuna [36].

TABLE I
HYPERPARAMETERS FOR THE PPO ALGORITHM AND THEIR CORRESPONDING VALUES

Hyperparameter	Value
Learning rate	0.001
Discount factor	0.99
Steps before update	2048
Clip range	0.2
Batch size	64

All training as well as remaining simulations were performed on a laptop PC with an Intel Core i5-10210U CPU, and an NVIDIA GeForce MX250 GPU.

F. Optimization-based Planner

To evaluate the upper limit of agent performance from the designed custom environment, an optimization-based planner was implemented in addition to the DRL agent. The optimization problem was defined as follows

$$\begin{aligned} \max_{a} \quad & R(s = s_0, a) \\ \text{where:} \quad & a = [y_{1:k-1}, v_{1:k-1}] \\ \text{s.t.} \quad & y_{\min} \leq y_i \leq y_{\max} \\ & v_{\min} \leq v_i \leq v_{\max} \end{aligned} \tag{21}$$

where a consists of the control points as defined in equation 12, R is the reward function given in equation 15, and $s = s_0$ is the initial state of the environment, given by equation 6, is known. The constraints on the vehicle velocities and lateral positions were the same as those defined for the DRL agent. In order to solve the above constrained non-linear optimization problem, the implementation of the Sequential Least SQuares Programming (SLSQP) algorithm from the SciPy library was used.

It is important to note that the trajectory generated by the optimization-based planner is not the best possible motion plan for the road profile, but given the proposed environment design along with the spline based motion profiles, it is a measure of the best performance the DRL agent can be expected to achieve.

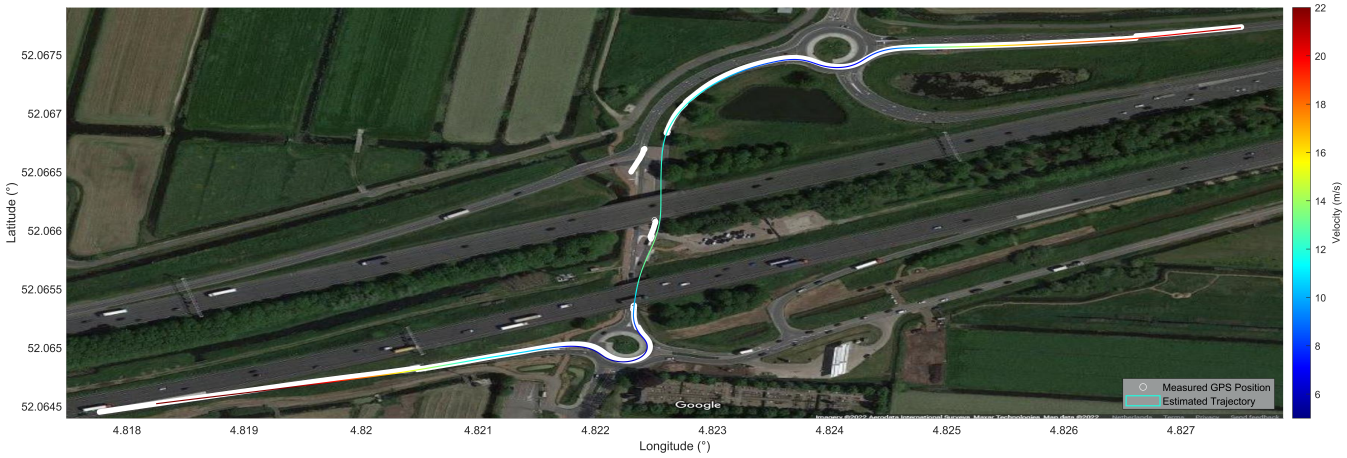

Fig. 3. A satellite view of the road section including the two roundabouts has been depicted. The vehicle positions as measured by the GPS sensor for one of the test runs have been shown in white, along with the estimated trajectory as obtained from the Kalman filter. As can be seen, the trajectory is reconstructed and the discontinuities in the data have been eliminated

III. Human Baseline Performance

In addition to comparing the performance of DRL agents with regards to targeting motion sickness, it is also interesting to compare the frequency content of the accelerations generated by DRL planner to the accelerations typically generated by human drivers. In this study, a human baseline performance was established by recording position, velocity and acceleration values as measured on a vehicle driven by volunteers on a chosen road section.

A road section in the Netherlands was chosen for the purpose of this driving performance evaluation. The test road begins at the exit ramp of motorway A12 (52.064°N, 4.818°E) and ends at the distributor road N420 (52.068°N, 4.828°E). The chosen path can be seen in Figure 3. In driving through the road section, the vehicle has to navigate through two roundabouts connected by a path with consecutive turns. The section was chosen as it consists of turns of different curvatures and lengths.

To evaluate the performance of the DRL agent, we focused only on the trajectory followed on each of the roundabouts, as the remaining portions of the path consisted mostly of long straight sections which would substantially increase the number of control points required, and subsequently the training time. The chosen sections of the roundabouts spanned 134 m in length, including the straight parts at entry and exit. The roundabouts will be referred to as RB1 and RB2 in sequential order respectively.

The motion profile as output by the agent was then evaluated in a high-fidelity IPG CarMaker environment with a multi-body vehicle model, as the custom training environment only used a point mass-model which would not be directly comparable to actual vehicle accelerations. The subsection III-A explains the experimental setup used, while subsection III-C details the controllers and vehicle model used for evaluating the motion planned by DRL agent.

A. Experimental Setup

For the purpose of the experiment, 6 volunteers were recruited, with ages ranging from 24 to 30 years, with an average age of 27.5. The driving experience among volunteers ranged from 4 to 10 years, with a mean of 7.2 years, so all volunteers can be considered fairly experienced. The volunteers were instructed to drive through the test route in a smooth manner while maintaining the highest pace to the best of their abilities. Each volunteer attempted two runs through the section, and the best run was chosen based on lowest interaction with traffic and minimum discomfort value. The trials were conducted during hours with minimal traffic to have the best representation of unobstructed human driving performance.

The chosen test vehicle was a Hyundai Tucson, equipped with an automatic transmission to avoid undesired accelerations from manual shifting. To record the vehicle trajectory, a high accuracy 100Hz Global Positioning System (GPS) was used in combination with an Inertial Measurement Unit (IMU). The experimental vehicle setup has been shown in Figure 4. The test runs resulted in a collection of vehicle position, velocity and acceleration profiles. The relevant portions of the motion profile in each of the roundabout sections were extracted using the position information from the GPS, and the acceleration values as recorded with the IMU were used to establish the human performance baseline as described in section IV.

B. Pre-processing Measurement Data

To establish the human baseline performance, the trajectory information from the isolated roundabout sections of the test scenario were collected. The measurements obtained from the GPS/GNSS sensor could not be used directly to ascertain the start and end of the trajectories due to errors resulting from measurement noise. To produce reasonable driving data representative of actual vehicle trajectories, a Kalman filter was implemented. A point mass kinematic model with state

Fig. 4. The test vehicle, a Hyundai Tucson (top), equipped with a double antenna GPS (bottom left) and IMU system (bottom right) for measuring driving performance of human drivers

given by vehicle position and velocity, and acceleration inputs was used as the model for system dynamics.

$$\begin{aligned}\mathbf{p}_k &= \mathbf{p}_{k-1} + \mathbf{v}_{k-1}\Delta t + \frac{1}{2}\mathbf{a}_{k-1}\Delta t^2 \\ \mathbf{v}_k &= \mathbf{v}_{k-1} + \mathbf{a}_{k-1}\Delta t\end{aligned} \tag{22}$$

where $\mathbf{p}_k$, $\mathbf{v}_k$ and $\mathbf{a}_k$ are the position, velocity and acceleration vectors in the global coordinate system at time step k and Δt is the sampling time. The state was taken as $\mathbf{x} = [\mathbf{p}^T\mathbf{v}^T]^T$. The acceleration vector was taken as the measured data from the IMU converted to global coordinates using the vehicle heading. The process noise was assumed to be a result of the noise in IMU measurements. The acceleration data was also low-pass filtered to remove high-frequency noise before being used.

The system was assumed to be completely observable with all state measurements $\mathbf{z}_k$ available from the GPS sensor.

$$\mathbf{z}_k = \begin{bmatrix}\mathbf{p} \\ \mathbf{v}\end{bmatrix} + \eta_k \quad \eta_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}) \tag{23}$$

Using the system dynamics given by equation 22 and the measurement model given by equation 23, and assuming normally distributed Gaussian noise for all sensors, the Kalman filter was implemented to obtain the estimated position and velocity profiles over the test road section. The desired portions of the trajectory about RB1 and RB2 were extracted using the estimated vehicle pose. An example of the measured and estimated trajectory and velocity profiles has been shown in Figure 3.

C. DRL Agent Performance Evaluation

The motion profiles generated by the DRL agent were based on a point-pass model, so in order to have a fair comparison with the human drivers, the trajectories were evaluated in a virtual IPG CarMaker environment. The vehicle model used was comparable in dimensions and kerb weight to the actual test vehicle in order to have as close a comparison as possible. To track the reference trajectories generated by the motion planner, a simple Stanley controller was implemented as follows

$$\delta = (\psi_r - \psi) + \text{atan}\frac{k_{steer}(y_r - y)}{v} \tag{24}$$

where δ and ψ are the steering input and heading of the vehicle respectively, y_r is the reference position, while k_{steer} is a parameter which decides the aggressiveness of the controller. The throttle percentage P_T or brake percentage P_B is decided as a weighted sum of reference forward acceleration $a_{x,r}$ with the error in velocity, scaled by a factor k_{drive} or k_{brake} depending on whether the vehicle is desired to be accelerated or decelerated.

$$P_T = k_{drive}(a_{x,r} + k_{speed}(v_r - v)) \times 100\% \tag{25}$$

$$P_B = k_{brake}(a_{x,r} + k_{speed}(v_r - v)) \times 100\% \tag{26}$$

IV. Results

The results have been divided into two subsections. Subsection IV-A goes into the frequency analysis of two DRL agents trained in a simple environment, one minimising motion sickness and the other optimizing general motion comfort described by total acceleration energy. The agents have also been compared to optimal planners. The subsection IV-B establishes the human baseline performance for the roundabout scenarios, and compares the performance of the trained DRL agent with the optimal planner as well as human drivers.

A. Frequency Domain Performance

For the purpose of investigating whether the DRL agent is able to target the low frequency acceleration component, a simple environment was used. The road length was assumed to be 100 m, with a single turn. The trajectory was defined with splines controlled by $k = 5$ control points. Two agents were trained in the same environment, with the only difference in their respective reward functions. Agent A was trained on a discomfort term calculated without frequency weighted accelerations, while agent B was trained using a reward function incorporating the band pass filters as described in section II-D. The accelerations outside the cut-off frequencies are attenuated by the band pass filters, and so the frequency weighted discomfort term is generally lower in value for comparable travel times. To compensate for this and ensure similar travel times for both agents, a weight of $W = 0.6$ and $W = 1$ was used for agent A and B respectively. Both agents were trained for 1M steps.

The planned trajectories of the agents A and B for a randomly generated scenario from the training environment have been shown in Figures 5 and 6 respectively. In the test case, the vehicle is initialised with a randomly generated speed of 7.25 m/s, and has to traverse through a sharp left turn.

As can be seen from the figures, both agents learn to accelerate in the straight sections of the road, and decelerate on approaching the corner. The spatial plans also are close

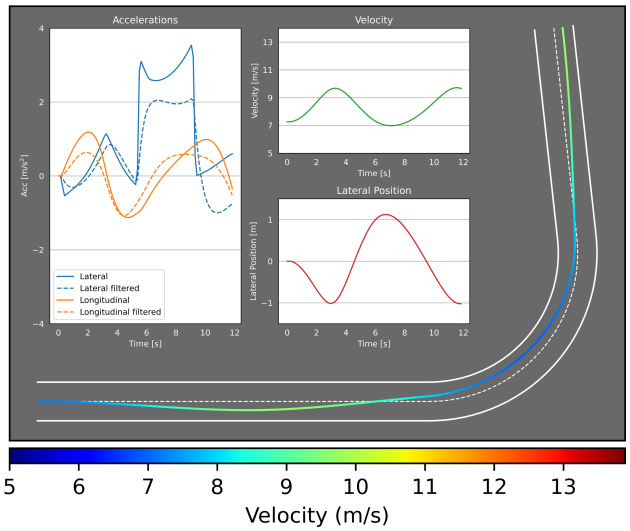

Fig. 5. Trajectory as planned by agent A. The subfigures depict the vehicle accelerations, velocity and positions from the motion plan. The entire path takes 11.84 s to navigate.

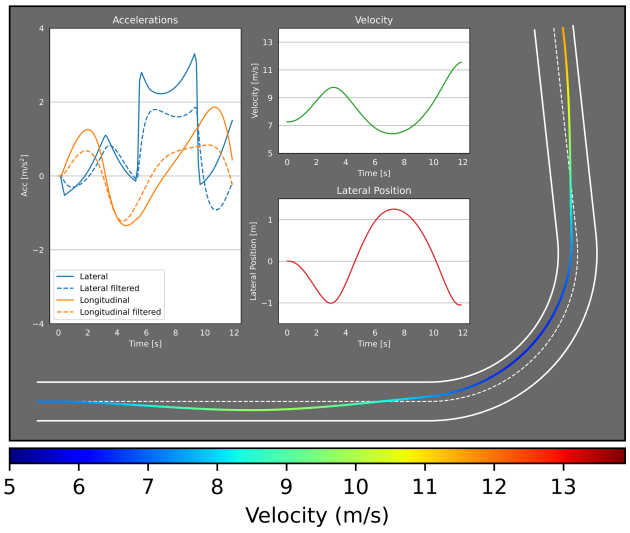

Fig. 6. Trajectory as planned by agent B. The subfigures depict the vehicle accelerations, velocity and positions from the motion plan. The entire path takes 11.87 s to navigate.

to a path that would be intuitively expected to be the most comfortable around the corner, with the vehicle entering from the outside edge, moving close to the apex and then exiting towards the outside edge of the corner. Both agents learn to utilize the complete limits of the available lateral deviation. In the particular test case shown here, the vehicle velocity range is not completely used, however, that is in the interest of producing lower vehicle accelerations. The peak lateral accelerations in both cases are around 3.5 m/s^2.

For the particular case being analysed, the frequency weighted discomfort term is 6.5% lower for the trajectory planed by agent B, with the same travel time as agent A. The drop in the discomfort term arises from the lower lateral

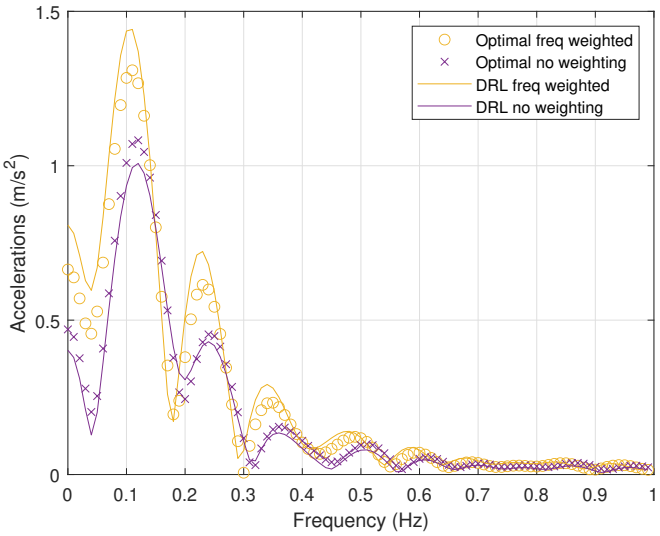

Fig. 7. Frequency content comparison of the longitudinal accelerations in the motion plans A and B. The line plots represent the DRL agents, while the scatter plots represent the optimal planners.

accelerations in motion plan generated by agent A. The travel time is unaffected due to the higher longitudinal accelerations, that is the vehicle brakes harder prior to the corner, and accelerates harder after exiting to reach a higher final velocity. Motion plan B has minimum and maximum velocities of 6.4 m/s and 11.5 m/s respectively, as opposed to 7.0 m/s and 9.7 m/s in motion plan A.

The preferential lowering of lateral accelerations by agent B can be attributed to the frequency filter. The longitudinal accelerations have a narrower band pass filter as opposed to lateral accelerations, and hence are attenuated more. In addition, the longitudinal frequency filter has a lower peak gain, to have the same area under the curve over the frequency range 0 to 1 Hz. The agent learns to increase accelerations beyond the cut-off frequencies, and target the frequencies of interest.

To analyze the frequency content of the accelerations, the Non-Uniform Fast Fourier Transform (NUFFT) has been shown in Figures 7 and 8. To further provide an insight into how the motion plans compare to ideal, the comparison with the optimization-based planner detailed in II-F has also been included. As expected from the acceleration values, it can be seen that the peak amplitudes of lateral accelerations are lower in motion plan B than A. Throughout the frequency band 0.0315 Hz to 0.25 Hz, the amplitudes are significantly lower in motion plan B. This does lead to higher peaks between 0.3 Hz to 0.9 Hz, but that is expected and desirable behaviour in our case. With DRL agent B, it can be seen that the energy is transferred from lateral to longitudinal accelerations, with significantly higher peaks compared to agent A. However, two important points need to be noted. Near the peak nauseogenic frequency of 0.2 Hz, motion plan B has a lower minima as compared to agent A. Also, lateral accelerations have significantly higher amplitudes throughout the relevant frequency spectrum as compared to longitudinal accelerations, and so the agent B learns to minimize low-frequency lateral accelerations

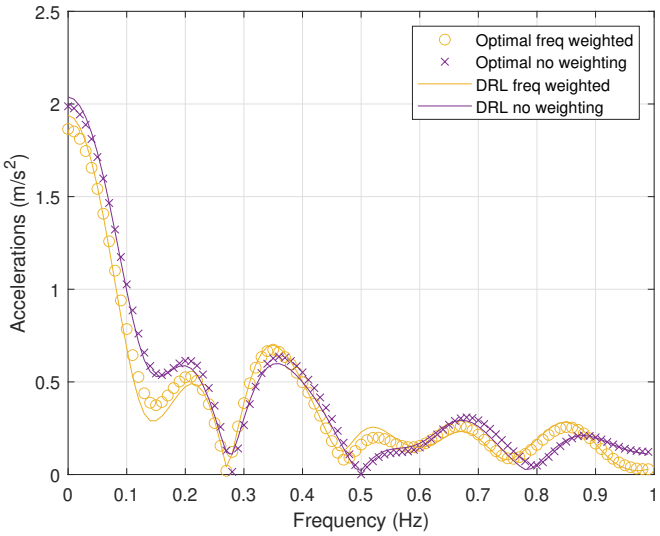

Fig. 8. Frequency content comparison of the lateral accelerations in the motion plans A and B. The purple and yellow line plots represent the DRL agents A and B respectively, while the scatter plots represent the optimal planners.

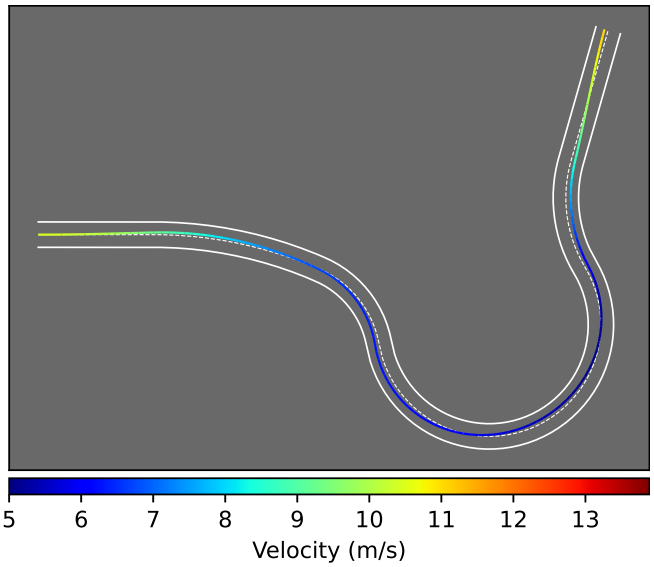

Fig. 9. Trajectory as planned by DRL agent for RB1. The weight on time for the agent is $W = 8$. The planned path takes 19.04 s to navigate.

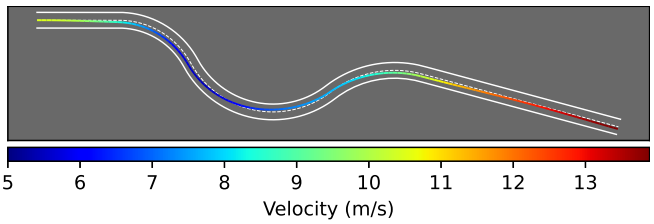

Fig. 10. Trajectory as planned by DRL agent for RB2. The weight on time for the agent is $W = 8$. The entire path takes 15.52 s to navigate.

at the cost of higher longitudinal accelerations. The study by Turner and Griffin [14] found higher correlation of lateral accelerations with motion sickness in passengers as compared to fore-aft accelerations, so the behaviour learnt by agent B works in this direction.

The above analysis was performed for a single randomly generated test case. However, to have a holistic idea of the performance of the agents, the average frequency weighted discomfort value and the travel times over 10,000 episodes were calculated. The frequency weighted discomfort value D for agent A was 19.61, while the for agent B was 17.73, which is a drop of 9.6%, quite significant for the relatively short and simple road profiles under consideration. The average travel times for the same were 9.78 s and 9.87 s respectively, which is a difference of less than 1%, and therefore can be considered comparable.

B. Comparison to Human Drivers

For comparison to human drivers, the DRL agent was trained on a 134 m road section with 6 sectors of varying lengths. Multiple agents were trained with varying weights on time ranging from $W = 4$ to $W = 16$, in order to have representative trajectories for different driving styles, and to study how the discomfort values depend on travel time. All agents were trained for 1.5M steps. The planned trajectories for our two evaluation scenarios RB1 and RB2, with a weight of $W = 8$ have been depicted in the Figures 9 and 10 respectively. As can be seen from the figures, roundabout 1 is a slower case with smaller radii of curvature, with a minimum radius 13 m. Roundabout 2 is a relatively faster corner with a minimum curvature radius of 17 m and only four curvature changes as opposed to five changes in roundabout 1.

The performance of the DRL agent has again been compared with the optimization-based planner described in section II-F. In addition, the nauseogenicity of planned motion has been compared with driving comfort of human drivers measured through the experimental setup as described in section III-A.

In order to have a representation of the performance of the motion plan over different weights, the discomfort values have been plotted in Figures 11 and 12. In both scenarios and with both planners, the planned discomfort values increase with a decrease in travel time, forming a Pareto front. It can be observed from the figures that in both cases, the discomfort values are higher with the DRL planner, which is as expected since it plans a sub-optimal path but with lower computation time. It is also evident from the plots that the discomfort values are significantly lower in RB2 due to the shorter travel time and lower curvatures of the turns. The data for both planners have been fit with a curve of the form

$$y = ax^b + c \tag{27}$$

With the trained agents, there is some inherent variability due to the randomness involved in the training process. To account for this variability, the fitted curves have been used for comparison. The performance of the DRL agent is between 10.9% to 12.9% of optimal over the range of weights $W = 4$ to $W = 16$ for RB1, and within 6.2% and 14.2% for RB2. The worse performance over RB2 is due to the increased

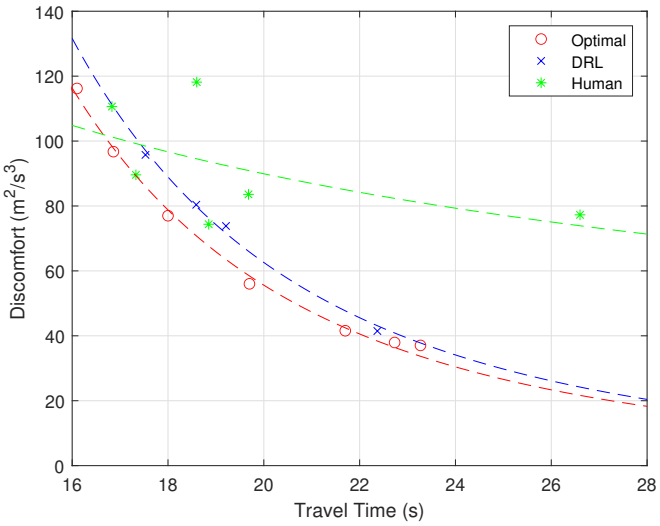

Fig. 11. Comparison of frequency weighted discomfort values and the travel times for the DRL agent and the optimal planner, for roundabout 1

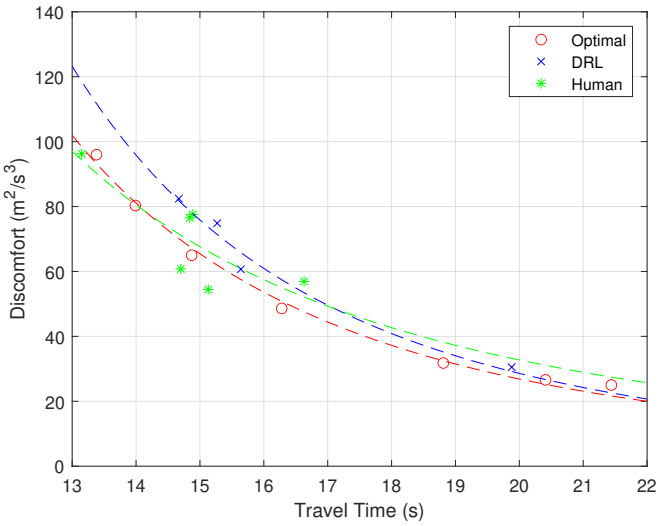

Fig. 12. Comparison of frequency weighted discomfort values and the travel times for the DRL agent and the optimal planner, for roundabout 2

difficulty of navigating the turn at higher speeds leading to higher accelerations, particularly at lower travel times.

The Figures 11 and 12 also depict the discomfort values as calculated for the human drivers, along with the fitted curves as per equation 27. For both roundabouts, it can be seen that there is significant variation among the performance of human drivers, with travel times ranging from 16.8 s to 26.6 s for RB1, and from 13.1 s to 16.6 s. The average travel times for RB1 and RB2 were 19.6 s and 14.9 s respectively. The frequency weighted discomfort ratings vary from 74.3 to 118.1 for RB1, with an average value of 92.3. For RB2, the corresponding range was from 54.4 to 96.3 with a mean value of 70.4. While there is a general trend as expected towards higher discomfort values with lower travel times, it can be seen that some drivers outperform others, clocking faster travel times with lower discomfort levels.

For the RB1 scenario, the DRL agents perform either on par or better than the human drivers with respect to frequency weighted discomfort, except for one driver. The best driver outperforms the DRL agent by 11.3%, with a discomfort rating close to the optimal for the environment. However, the DRL agent significantly outperforms the worst driver, by a margin of over 200%. It can be seen that the discomfort rating of the DRL agents show a sharper increasing trend than the human drivers with lower travel times, and this can be attributed to the larger modelling errors associated with the point mass model on approaching higher vehicle speeds and acceleration values.

The RB2 scenario, on the other hand does highlight some limitations in the design of the custom training environment itself. 3 human drivers drive along trajectories with comparable nauseogenicity to the DRL agents, exhibiting discomfort values within 5% of the DRL agent. The remaining 3 drivers drive along trajectories with much lower discomfort values, even improving upon the minimum discomfort levels obtainable from the environment. The reason for this, as mentioned earlier can be attributed to a combination of two factors. The first reason is the constraints of the motion profile itself due to the use of splines for planning the motion profiles. The splines reduce the action space of the agent and therefore keep training times to practically achievable values, but also introduce a cap on the best performance achievable by the agent. The second reason could be the higher modelling errors with using a point-mass model for trajectories, but this effect will only be pronounced with faster travel times and higher acceleration values.

Another comparison of interest is the trend of lateral and longitudinal accelerations with varying the weights, which gives further insight into the time-comfort compromise of the planner. The plots have been depicted in Figure 13. The values of overall RMS acceleration and lateral accelerations are as expected, showing an increasing trend with higher values of W. The overall accelerations are lower than the optimal at $W = 16$, as at that weight the agent learnt to predict slower trajectories leading to lower RMS accelerations. An interesting insight from the graphs is the low increase (and even reduction from $W = 12$ to $W = 16$) of the longitudinal accelerations with increased weight, both with DRL and the optimal planner. With increasing weights W on time the planner tolerates a much higher lateral acceleration value in interest of travel time, leading to only a small increase in longitudinal accelerations.

The metric in which the DRL agent comprehensively outperforms the optimization-based planner is the computational time. Although the DRL agent predicts sub-optimal trajectories with discomfort values 10 to 15% higher than optimal, the trained agent once loaded only takes between 1-2 ms to predict the motion plan for the given road profile. In comparison, the optimal planner takes an average of 5 s to compute the optimal motion plan. This is a significant improvement in computation time, for a sub-optimal but relatively 'good' motion profile. This result is further strengthened by the fact that the discomfort ratings of the DRL trajectories are comparable to those achieved by human drivers, particularly at lower vehicle velocities.

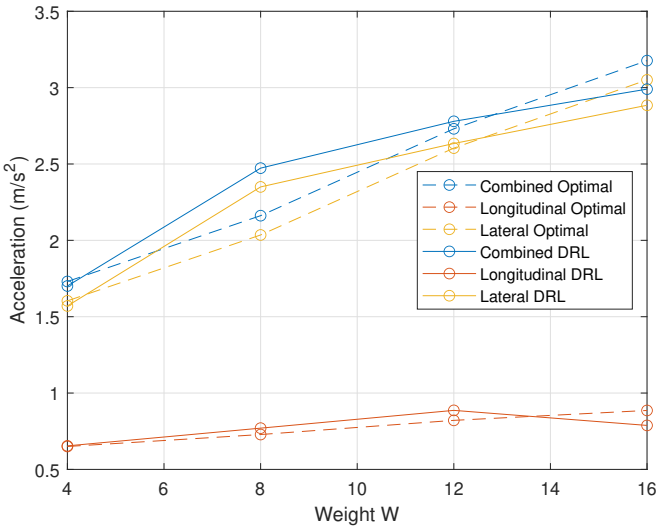

Fig. 13. Change in RMS acceleration values with varying weight W

V. Conclusions

A. Contributions

In this research, a novel method of minimizing motion sickness in motion planning through the use of Deep Reinforcement Learning has been presented. The naseogenicity of planned trajectories was evaluated by using a discomfort term, which used acceleration values passed through frequency weighting filters, derived from motion sickness models present in literature. It was shown that the use of the discomfort term in the reward function of the DRL agent allowed it to successfully learn to target frequencies of interest which are primarily responsible for motion sickness. The frequency weighted discomfort values were found to be 9.6% lower on average as compared to an agent trained with a discomfort term calculated using unweighted accelerations. The difference arises due to shifting of energy from lateral to longitudinal accelerations, and to frequencies beyond the band of interest.

The DRL agent was evaluated on two roundabout sections modelled on a real road in the Netherlands. In order to have a representation of DRL agents with different driving styles, multiple agents were trained with varying weights $W \in [4, 16]$. The performance was compared to an optimal planner, as well as a human performance baseline established by performing trials with volunteers. The average discomfort rating values over two roundabout sections were found to be 92.3 and 70.4 respectively, with average travel times of 19.6 s and 14.9 s. It was found that the DRL agents provided comparable discomfort values to the human drivers, particularly around the slower roundabout. The measured discomfort was however, higher than the optimization-based planner for all different driving styles, usually in a range of 10-15%. The DRL agent did offer massive improvements in computation time, by reducing the online computation time by three orders of magnitude.

B. Limitations and Future Work

It was shown that for scenarios with a reasonable complexity level, DRL can be used for motion planning in a manner which reduces acceleration frequencies in nauseogenic bands. The performance of the agent is however limited by the design of the environment, and to use more complex environments, agents with a larger state and action space need to be trained. It also needs to be noted that for real roads with more complex profiles, such as multi-lane highways, highway ramps or intersections, the performance of the agent still needs to be proven. The state representation of the agent itself is not general enough to be applied to all different kinds of road lengths and curvatures which may be encountered. The use of Recurrent Neural Networks (RNNs) could be investigated for incorporating a variable state space so as to have a more general representation of the road profile.

The challenges generally encountered with DRL are also present in this study. The agents require a long time to train, and training time increases exponentially along with the dimensionality of state and action space. The motion plans generated, although substantially reduce on-line computation time, they produce sub-optimal results.

In addition, it is also not clear whether the objective improvements in motion sickness dose values would translate to real world comfort improvements for passengers. It may very well be the case that focusing on motion sickness mitigation might lead to reduction in general comfort of the passenger, and the benefits of the motion plan may only be appreciable with longer journeys and curvy roads which promote low frequency accelerations. The perceived benefits of the proposed motion plan can be evaluated through driving simulator or on-road experiments with human subjects.

References

[1] B. Pfleging, M. Rang, and N. Broy, "Investigating user needs for non-driving-related activities during automated driving," 12 2016, pp. 91–99.
[2] C. Diels and J. Bos, "Self-driving carsickness," *Applied ergonomics*, vol. 53, 10 2015.
[3] J. Reason and J. J. Brand, *Motion sickness / J. T. Reason, J. J. Brand*. Academic Press London ; New York, 1975.
[4] A. Rolnick and R. Lubow, "Why is the driver rarely motion sick—the role of controllability in motion sickness," *Ergonomics*, vol. 34, pp. 867–79, 08 1991.
[5] G. Bertolini and D. Straumann, "Moving in a moving world: A review on vestibular motion sickness," *Frontiers in Neurology*, vol. 7, 2016.
[6] J. O'Hanlon and M. McCauley, "Motion sickness incidence as a function of vertical sinusoidal motion," *Aerospace medicine*, vol. 45, pp. 366–9, 05 1974.
[7] M. Griffin, "7 - motion sickness," in *Handbook of Human Vibration*. London: Academic Press, 1990, pp. 271–332.
[8] J. Golding, A. Mueller, and M. Gresty, "A motion sickness maximum around the 0.2 hz frequency range of horizontal translational oscillation," *Aviation, space, and environmental medicine*, vol. 72, pp. 188–92, 04 2001.
[9] B. Donohew and M. Griffin, "Motion sickness: Effect of the frequency of lateral oscillation," *Aviation, space, and environmental medicine*, vol. 75, pp. 649–56, 09 2004.
[10] S. Salter, S. Kanarachos, C. Diels, and C. D. Thake, "Motion sickness in automated vehicles with forward and rearward facing seating orientations," *Applied Ergonomics*, vol. 78, pp. 54–61, 02 2019.
[11] O. Kuiper, J. Bos, C. Diels, and E. Schmidt, "Knowing what's coming: Anticipatory audio cues can mitigate motion sickness," *Applied ergonomics*, vol. 85, p. 103068, 05 2020.

[12] M. Miksch, M. Steiner, M. Miksch, and A. Meschtscherjakov, "Motion sickness prevention system (msps): Reading between the lines," in *Adjunct Proceedings of the 8th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, 10 2016, p. 147–152.

[13] R. Koppa and G. Hayes, "Driver inputs during emergency or extreme vehicle maneuvers," *Human Factors*, vol. 18, pp. 361–370, 08 1976.

[14] M. Turner and M. J. Griffin, "Motion sickness in public road transport: The relative importance of motion, vision and individual differences," *British Journal of Psychology*, vol. 90, no. 4, pp. 519–530, 1999.

[15] L. Labakhua, U. Nunes, R. Rodrigues, and F. Leite, "Smooth trajectory planning for fully automated passengers vehicles: Spline and clothoid based methods and its simulation," in *Lecture Notes in Electrical Engineering*, vol. 15, 01 2006, pp. 89–96.

[16] M. McNaughton, C. Urmson, J. M. Dolan, and J.-W. Lee, "Motion planning for autonomous driving with a conformal spatiotemporal lattice," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 4889–4895.

[17] S. Gim, L. Adouane, S. Lee, and J.-P. Dérutin, "Clothoids composition method for smooth path generation of car-like vehicle navigation," *Journal of Intelligent & Robotic Systems*, vol. 88, 10 2017.

[18] R. Lattarulo, E. Martí, M. Marcano, J. Matute, and J. Pérez, "A speed planner approach based on bézier curves using vehicle dynamic constrains and passengers comfort," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2018, pp. 1–5.

[19] Y. Zheng, B. Shyrokau, and T. Keviczky, "Comfort and time efficiency: A roundabout case study," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 3877–3883.

[20] ——, "3dop: Comfort-oriented motion planning for automated vehicles with active suspensions," in *2022 IEEE Intelligent Vehicles Symposium (IV)*, 2022, pp. 390–395.

[21] Z. Htike, G. Papaioannou, E. Velenis, and S. Longo, "Motion planning of self-driving vehicles for motion sickness minimisation," in *2020 European Control Conference (ECC)*, 2020, pp. 1719–1724.

[22] D. Li and J. Hu, "Mitigating motion sickness in automated vehicles with frequency-shaping approach to motion planning," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7714–7720, 2021.

[23] R. Ukita, Y. Okafuji, and T. Wada, "A simulation study on lane-change control of automated vehicles to reduce motion sickness based on a computational mode," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2020, pp. 1745–1750.

[24] S. Aradi, "Survey of deep reinforcement learning for motion planning of autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 740–759, 2022.

[25] A. E. Sallab, M. Abdou, E. Perot, and S. Yogamani, "End-to-end deep reinforcement learning for lane keeping assist," 2016.

[26] P. Wolf, K. Kurzer, T. Wingert, F. Kuhnt, and J. M. Zollner, "Adaptive behavior generation for autonomous driving using deep reinforcement learning with compact semantic states," *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018.

[27] C.-J. Hoel, K. Wolff, and L. Laine, "Automated speed and lane change decision making using deep reinforcement learning," *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018.

[28] P. Wang and C.-Y. Chan, "Formulation of deep reinforcement learning architecture toward autonomous driving for on-ramp merge," *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, 10 2017.

[29] Á. Fehér, S. Aradi, F. Hegedüs, T. Bécsi, and P. Gáspár, "Hybrid ddpg approach for vehicle motion planning," in *Proceedings of the 16th International Conference on Informatics in Control, Automation and Robotics, ICINCO 2019, Vol. 1*, 2019, pp. 422–429.

[30] C.-J. Hoel, K. Driggs-Campbell, K. Wolff, L. Laine, and M. J. Kochenderfer, "Combining planning and deep reinforcement learning in tactical decision making for autonomous driving," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 2, p. 294–305, 2020.

[31] P. Wang and C.-Y. Chan, "Autonomous ramp merge maneuver based on reinforcement learning with continuous action space," 2018.

[32] C. Paxton, V. Raman, G. D. Hager, and M. Kobilarov, "Combining neural networks and tree search for task and motion planning in challenging environments," 2017.

[33] R. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, pp. 229–256, 05 1992.

[34] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 07 2017.

[35] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, 01 2021.

[36] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

3

Results and Discussions

This chapter consists of two sections. The first section details some additional results to supplement those provided in the journal paper given in chapter 2. Further, a detailed discussion on the issues faced with the method as well as possible solutions to overcome these issues have been provided in section 3.2.

3.1. Additional Results

Figures 3.1 and 3.2 are the collected fourier transforms of respectively the longitudinal and lateral accelerations generated by all the human drivers as well as the DRL agents of different weights W. Figures 3.3 and 3.4 depict the corresponding information for RB2.

As defined in Appendix C, the peak nauseogenic frequency band for longitudinal accelerations is between 0.15Hz and 0.25Hz. It can be seen from Figures 3.1 and 3.3 that peak longitudinal accelerations in the trajectories generated by human drivers are concentrated in the frequency range between 0.1Hz and 0.2Hz, nearly coinciding with the peak nauseogenic frequencies. With the DRL motion plans, however, the peak is shifted to lower frequencies, with the highest amplitudes between 0 to 0.1Hz. Another interesting observation is that the high frequency content is also significantly lower in the longitudinal accelerations from the DRL motion plans, which points to higher longitudinal jerks from human drivers.

With lateral accelerations, the band of interest is a broader one, between 0.0315Hz to 0.25Hz. Here again it can be seen that the peaks are consistently lower with the DRL motion plans in the low frequency spectrum. To account for the difference in travel times, even with the fastest human driver acceleration profile removed, the peaks in the low frequency band remain higher than the DRL agents. However, with lateral accelerations, the amplitudes are much higher in the high frequency 0.5 Hz to 1 Hz range for the DRL motion plans with peak amplitudes higher than 1m/s^2 compared to peak amplitudes of 0.2m/s^2 for human drivers. It can be deduced that this shift of lateral accelerations from the low frequency to high frequency region leads to lower nauseogenicity of the DRL motion plans as compared the human drivers. However, it should be noted that these high frequency accelerations can cause higher immediate discomfort to passengers due to more jerk, as has been discussed in more detail in the subsequent section.

3.2. Discussion

Over the course of designing the training environment as well as during the evaluation of the DRL agorithm, several issues were encountered. This section will go into the details of the issues encountered, as well as some possible measures to tackle them.

3.2.1. Environment Design

The current environment design makes use of spline based velocity and position planning. This limits the action space of the DRL agent to the number of control points used to define the splines. This was

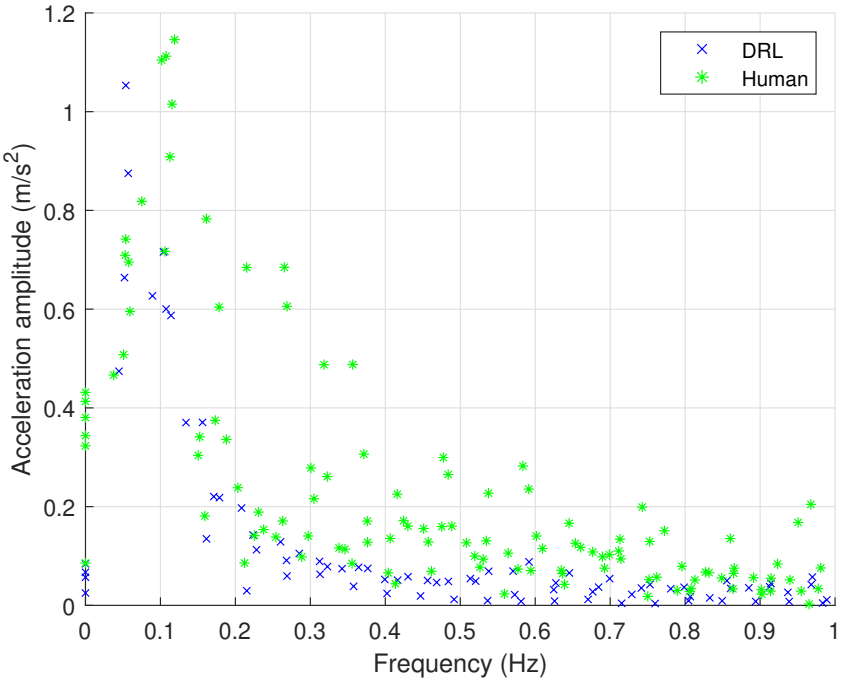

Figure 3.1: Frequency content comparison of the longitudinal accelerations of all human drivers with DRL agents for RB1.

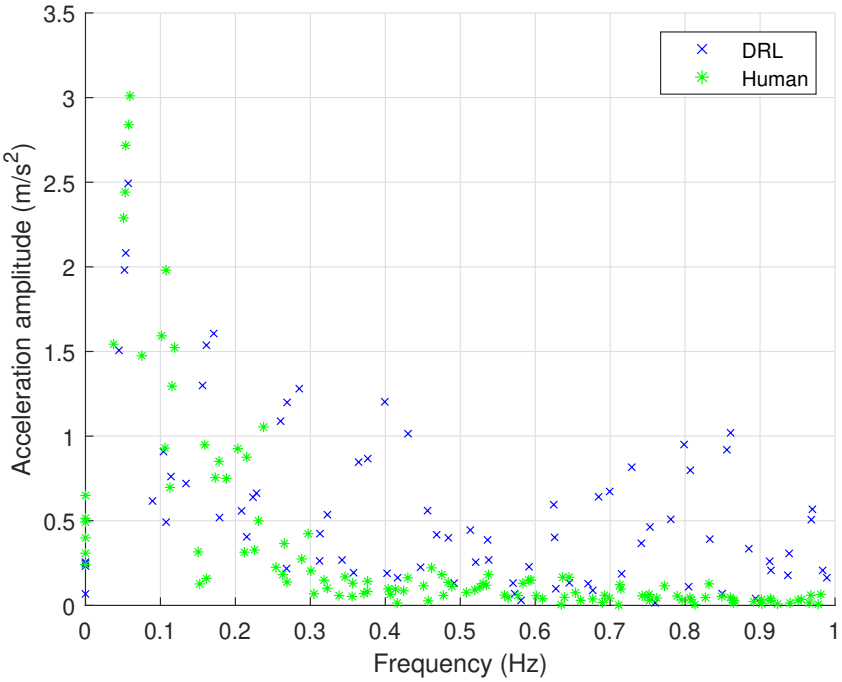

Figure 3.2: Frequency content comparison of the lateral accelerations of all human drivers with DRL agents for RB1

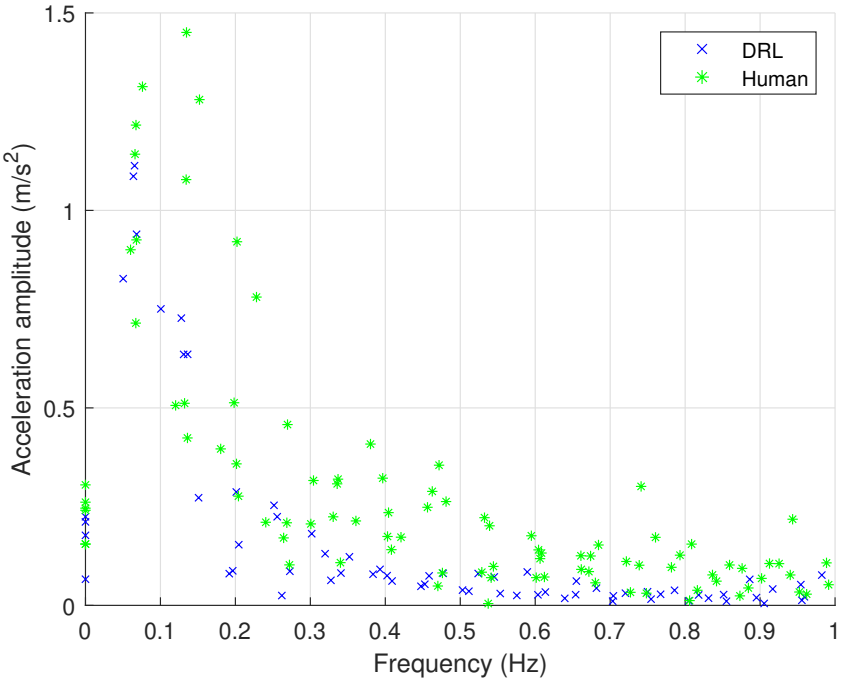

Figure 3.3: Frequency content comparison of the longitudinal accelerations of all human drivers with DRL agents for RB2

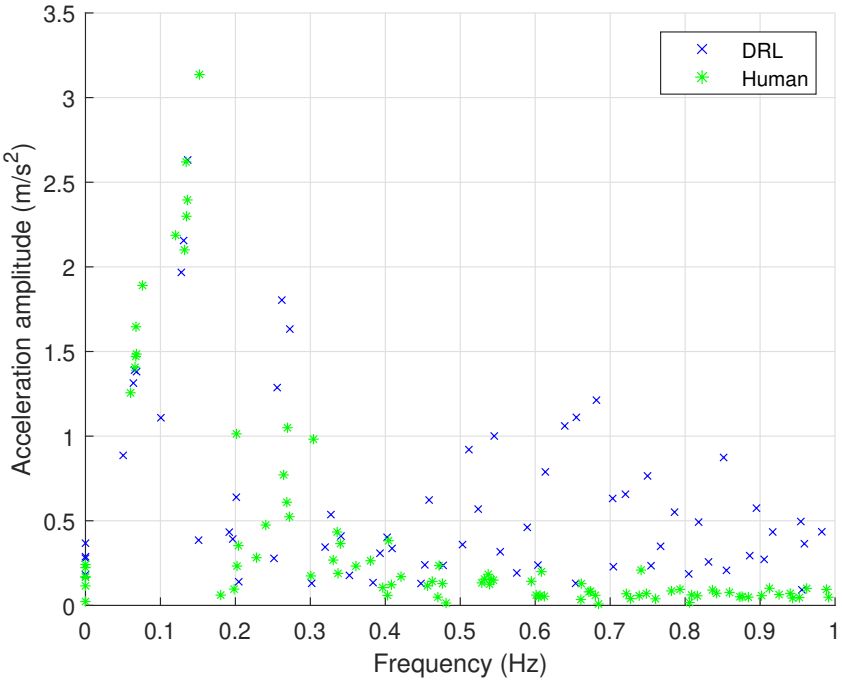

Figure 3.4: Frequency content comparison of the lateral accelerations of all human drivers with DRL agents for RB2

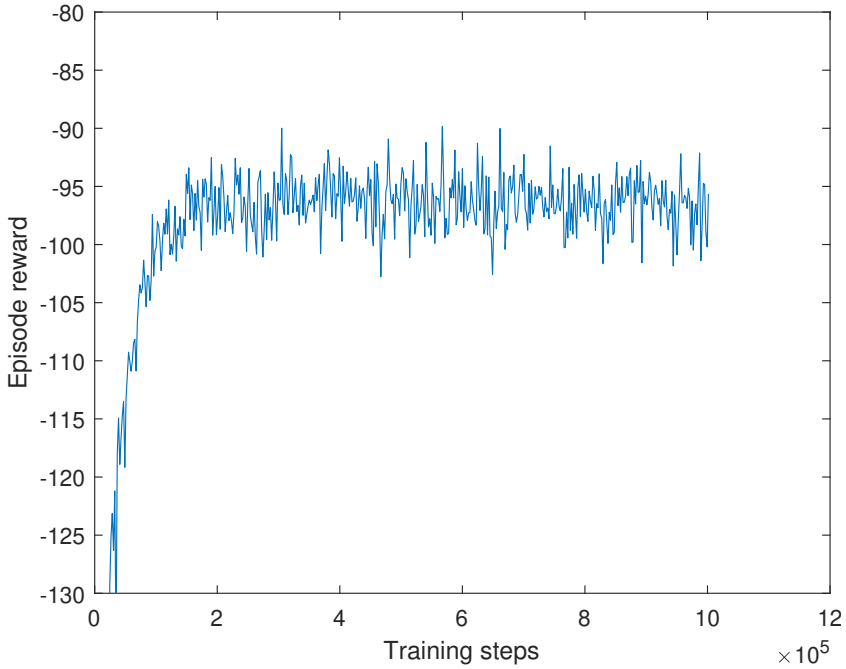

Figure 3.5: Mean episode rewards as a function of training time for DRL agent with action space of size 8. It can be seen that the rewards stabilize after 400-500k steps, and no further increase is observed.

done with the interest of maintaining continuity in the planned accelerations, and to limit the size of the action space. However, this limits the degree of freedom available to the agent to plan the most optimal path possible, and therefore enforces an upper ceiling on the real world performance. This could be observed from the performance comparison with human drivers for RB2, as detailed in the journal paper.

In order to remedy this problem, the easiest way is to increase the number of control points for the agent to plan. In the thesis, splines with 5 and 8 control points were used for two different types of DRL agents trained, with corresponding action spaces of 8 and 14 respectively (for the combined position and velocity splines). For a Fully Connected NN with 64x64 neurons in the hidden layer, an increase in the action space by n actions will lead to $64n$ more weights in the NN to be trained. As was observed from training the DRL agent in the two environments with action space of 8 and 14 respectively, the number of episodes required for the reward to stabilize increased from 500,000 to 1.5M steps. This can be seen in figures 3.5 and 3.6. Training for 1.5M steps already required a wall clock time of 16 hours with the available computational resources, and therefore larger action spaces were not explored. However, it can be extrapolated from the performance on simpler environments that increasing the action space size and providing more control points for the agent to plan the trajectory can result in improved performance, provided long enough training times are maintained. Another possible solution could be to use more control points only for planning the position while maintaining a minimum number of control points for velocity planning, as the lateral acceleration values are of greater importance for our purpose of motion sickness mitigation.

Another aspect of the environment design which can be improved upon is the design of the state space itself, which consists primarily of the road profile information. Currently the state space as implemented in this thesis is as defined in section B.1.1, and has the road information in terms of the sector lengths and curvatures. The maximum length of road profile that the agent can plan for is fixed, and so is the maximum number of sectors with changing curvature values. While the road length and curvature limits are defined such that they cover a wide range of possible cases that the agent may encounter in the real world, there might still be curvature-length combinations which the agent is not equipped to handle.

The state space could be modified to handle a wider range of possible road profiles by two possible methods. The size of the state space itself can be increased, thus increasing the maximum number

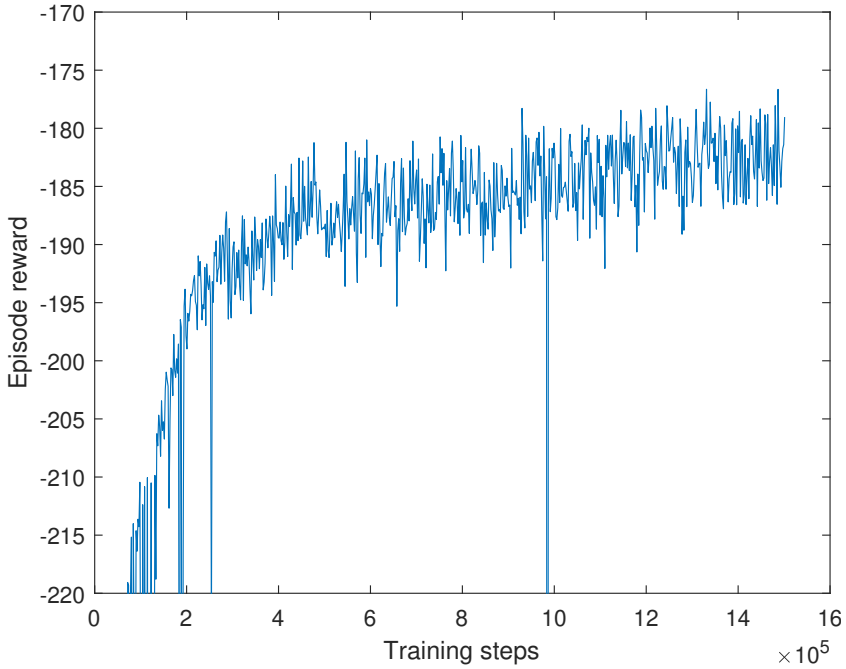

Figure 3.6: Mean episode rewards as a function of training time for DRL agent with action space of size 14. The rewards are steadily increasing throughout the training period of 1.5M steps

of curvature and length combinations it can handle. Shorter roads can then be accounted for by using zero-padding for the states which remain unused. This is not a very elegant solution, and may lead to increased training times, similar to that encountered with an increase in action space size.

A possible method to incorporate varying number of road sectors could be to take inspiration from natural language processing problems which make use of Recurrent Neural Networks to handle varying number of input states [31]. They would provide the added advantage of being able to learn temporal dependencies on the previous states, which could be particularly attractive for identifying low frequency acceleration effects.

3.2.2. Evaluation of Real-World Benefits

It has been shown that the DRL motion planner shows good objective performance levels, with frequency-weighted discomfort values comparable to those of human drivers. However, the nature of the accelerations generated from the motion plan are quite different to those of human drivers, as elaborated in section 3.1. The focus on reducing low-frequency accelerations will inevitably lead to higher acceleration energy being concentrated elsewhere in the frequency spectrum, particularly at higher frequencies as is evidenced in figure 3.2. This would translate to higher jerk levels for passengers, which could be a source of discomfort. Motion sickness generally manifests over longer journeys, and for shorter journeys, high jerks could overshadow the anticipated benefits from the motion plan. These effects therefore need to be evaluated and quantified through subjective evaluations of the motion plans by human volunteers, to give a realistic picture of how the proposed motion plans translate to the real world.

3.2.3. DRL Issues

DRL is still a relatively new field of research with numerous open challenges such as reproducibility, hyperparameter tuning, instability in training and so on [32]. Multiple challenges relating to DRL were faced over the course of this research as well. Hyperparameter tuning requires search over a very large possible space due to the number of parameters to be optimized, and due to long training times, the search itself is quite time consuming. Any modifications to the environment need to be again followed by hyperparameter tuning. Occasional instabilities in training were encountered, as can be seen from Figure 3.6 around 1m steps, but PPO in general exhibited reliable performance. Constraint handling is

another aspect which needs to be further evaluated, in particular with respect to incorporating obstacle avoidance and speed limits into the motion planner. DRL cannot deal with explicit constraints, but this can be achieved indirectly through enforcing penalties on exceeding said constraints. Another common way to circumnavigate this issue is by using DRL in combination with a high-level rule based planner which enforces constraints on top of the planned path as a safety net [33].

The most significant problem was the reproducibility of results. The learning process in the DRL algorithms involve stochasticity, and in our particular case, the generation of the training environment itself is random. This leads to high variability between runs, which inevitably causes reproducibility to suffer. This problem is somewhat mitigated through the maximum possible reduction of dimensionality of state and action space, which in our case was achieved through the use of spline based planning.

4

Conclusions and Future Work

4.1. Conclusions

In this thesis, a Deep Reinforcement Learning approach to motion sickness mitigation in the motion planning layer of automated vehicles has been developed. The method is centred around a reward function which uses frequency weighted accelerations as a measure of discomfort, and aims to shape the frequency response of planned accelerations of the agent.

The frequency shaping effect of the reward function was evaluated by comparison with a DRL agent trained on unweighted accelerations. Over 10,000 episodes, it was found that the developed DRL agent learnt to reduce frequency weighted discomfort by nearly 10%, while maintaining comparable travel times as the benchmark agent. Both agents were shown to learn near optimal frequency response for their respective reward and environment definitions.

An important contribution of the thesis was the evaluation of the discomfort levels of the trained DRL agent in a realistic simulation environment, and then subsequent comparison to driving comfort performance of human drivers. The vehicle trajectories and accelerations for 6 drivers navigating through two roundabout sections were collected, and used as a benchmark to evaluate DRL performance. The DRL agent outperformed the human drivers in a slower roundabout section, but was observed to have higher discomfort levels in the other roundabout section. The frequency spectrum comparison revealed that the DRL agent led to targeted reductions in accelerations in the undesirable frequency band, while having higher amplitudes in the remaining portions of the frequency spectrum.

The DRL agent was shown to have drastically lower computational times than the optimization-based planner in the same environment, while generating discomfort values 10 to 15% worse. In conclusion, the thesis shows that the developed DRL algorithm is a viable alternative to classical optimization-based techniques for motion planning, and has the potential to be applied for the purpose of motion sickness minimization.

4.2. Future Work

The DRL framework developed in this thesis is a first step in its application towards motion sickness mitigation. However, as discussed in chapter 3, it does suffer from limitations, and there is significant scope for improvement in order to create a planner which can be implemented for a wide range of problems. The improvements can be classified into three directions: design of the environment, evaluation of real-world benefits, and improvements in the DRL algorithm itself.

The applicability to more complex environments with larger action spaces and more general state spaces can be investigated, albeit keeping in mind the higher computational resources required for training. The subjective improvements in motion comfort corresponding to the objective results need to be verified through either on-road or driving simulator experiments with human volunteers. Finally, incorporation of constraints in the motion plan such as speed limits, and obstacle avoidance can be investigated by adding penalties in the reward function.

A

Deep Reinforcement Learning

Reinforcement Learning (RL) is a sub-field of Machine Learning, which lies between the supervised and unsupervised learning paradigms. RL takes inspiration from learning in humans and animals, by trying to teach behaviours to an agent through the process of trial and error. The desirable outcomes are rewarded, and undesirable ones are penalized. The objective of an RL agent is to maximise long term cumulative rewards through learning an optimal policy. It does not require any training data to learn, but does require a training environment with which it interacts and which provides a reward corresponding to the actions of the agent.

Apart from the mentioned advantage of not requiring training data, RL also does not require a model of the system to learn a satisfactory control policy. This allows it to be applied to tasks where the system is too complex to be modelled mathematically. Similar to other ML algorithms, RL also requires minimal online computation resources, with the drawback of long training times, especially with more complex tasks.

In this appendix, the central concepts of DRL have been explained, followed by a overview of the popular DRL methods and the method used in this study.

A.1. Basic Concepts

The RL problem is typically represented as a Markov Decision Process (MDP), which consists of a tuple $< S, A, P, R >$. Here S is the set of all possible states s, and A is the set of feasible actions a, also known as the state and action spaces respectively. P is the probability transition function given by

$$s_{k+1} \sim P(.|s_k, a_k) \tag{A.1}$$

where $s_k \in S$ and $a_k \in A$ are the states and actions at time step k. R is the reward function which is given by

$$r_k = R(s_k, a_k) \tag{A.2}$$

where r_k is a scalar value, and is termed the reward. The most important property of a Markov process is that the complete information of the environment is contained within the state at the current time, and the environment does not depend on either past or future states.

The RL agent is given by a policy π, a mapping from the state space to the action space, which defines the actions the agent takes in any given state. The policy in case of DRL, is represented in the form of a Neural Network (NN) parameterised by θ. This can be represented as the following equation

$$a_k \sim \pi_\theta(s_k) \tag{A.3}$$

The actions a_k might be discrete or continuous. Th agent is initialised with a state s_0, and it samples actions a_k from its policy, leading to successive states as generated by the state transition function P

till it reaches a terminal state. This sequence of states and actions is called a trajectory τ, also known as an episode.

$$\tau = (s_0, a_0, s_1, a_1, \dots s_H) \tag{A.4}$$

H in the above equation is the horizon, and s_H is the terminal state.

The objective of the agent is to maximise cumulative reward over a period of time, known as the return R. The return at time k is given as

$$R(\tau) = \sum_{k=0}^{\infty} \gamma^k r_k \tag{A.5}$$

where $\gamma \in [0, 1]$ is a discount factor, which ensures that the value of the return R remains bounded in the infinite sum. It also weighs the immediate rewards more than those obtained in the distant future. Higher discount factors lead to higher impact of rewards in the distant future on the return. This causes the agent to learn behaviours which are more far-sighted, and improve long term returns. However, too high discount factors may lead the agent to correlate actions with rewards too far into the future, which were not a result of their contribution.

The objective J of the DRL agent can be formally defined as

$$\max_{\pi} J(\pi) = \mathop{\mathbb{E}}_{\tau \sim \pi} [R(\tau)] \tag{A.6}$$

The agent tries to maximise the objective J, which is the expectation of the return R by learning an optimal policy π^* which can be expressed as

$$\pi^* = \arg\max_{\pi} J(\pi) \tag{A.7}$$

To maximise the objective function, there are numerous algorithms in literature, with the most common being value based and policy based methods. The following sections give a brief overview of value based and policy based methods.

A.2. Value Approximation Based Algorithms

A concept very central to RL is the value function. The value function V^π expresses the value of being in a particular state s, and is expressed as the expected value of the return R from the state on following a policy π.

$$V^\pi(s) = \mathop{\mathbb{E}}_{\tau \sim \pi} [R(s_0 = s)] \tag{A.8}$$

The higher the value V^π of a state, the more the possibility of obtaining a high return from the state on following the policy π. Another related function is the action-value function or the Q-function, which focuses also on the value of the initial action a taken in any state. The Q-function is defined as

$$Q^\pi(s, a) = \mathop{\mathbb{E}}_{\tau \sim \pi} [R(s_0 = s, a_0 = a)] \tag{A.9}$$

To calculate the Q-value for a state-action pair (s, a), the first action taken in the state s is a, with the remaining trajectory sampled from the policy π. As opposed to the value function, the Q-function separates the advantage of taking a specific action in the given state.

The optimal Q-function Q^*, if known, can be used to easily find the optimal action, the action which maximises the objective J for a given state

$$a^*(s) = \arg\max_{a} Q^*(s, a) \tag{A.10}$$

Another function used to quantify the value of taking an action in a state as compared to other actions is the advantage function A^π. The advantage function is calculated as follows

$$A^\pi(s, a) = Q^\pi(s, a) - V^\pi(s) \tag{A.11}$$

The advantage function is used to find the relative goodness of an action as compared to other actions, as it separates the additional value of taking an action, from only the value of the state.

Many DRL algorithms such as Deep Q-learning (DQN) [34], Double DQN [35] and Duelling DQN [36] try to approximate the optimal policy π^* by estimating the optimal Q-function $Q^*(s, a)$. These methods are referred to as value approximation based methods.

A.3. Policy Based Algorithms

A contrasting approach to value based methods are policy gradient-based methods, which try to optimise the policy directly rather than approximating the optimal value function. Policy-based algorithms work by trying to optimize the policy directly through means of gradient ascent on the parameters θ

$$\theta_{k+1} = \theta_k + \alpha \nabla_\theta J(\pi_\theta)|_{\theta_k} \tag{A.12}$$

where α is a learning rate, and θ are the parameters of the policy network. The objective function J as defined in equation A.6, is the expected value of the return over all trajectories $\tau \in \mathbb{T}$, and can be rewritten in terms of the probability distribution of trajectories $p_\theta(\tau)$ and the rewards $r(\tau)$ as follows

$$\max_\pi J(\pi) = \mathop{\mathbb{E}}_{\tau \sim \pi}[R(\tau)] \tag{A.13}$$

$$= \int_{\mathbb{T}} p_\theta(\tau) r(\tau) d\tau \tag{A.14}$$

$$\tag{A.15}$$

Then differentiating the above equation, the gradient of the objective J can be calculated as follows

$$\nabla_\theta J(\theta) = \int_{\mathbb{T}} \nabla_\theta p_\theta(\tau) r(\tau) d\tau \tag{A.16}$$

$$= \int_{\mathbb{T}} p_\theta(\tau) \nabla_\theta \log p_\theta(\tau) r(\tau) d\tau \tag{A.17}$$

$$= \mathbb{E}\{\nabla_\theta \log p_\theta(\tau) r(\tau)\} \tag{A.18}$$

The probability distribution $p_\theta(\tau)$ can be expressed in terms of the probability distribution of the initial state, probabilities of the state transition function and the policy itself

$$p_\theta(\tau) = p(s_0) \Pi_{k=0}^{H} p(s_{k+1}|s_k, a_k) \pi_\theta(a_k|s_k) \tag{A.19}$$

The usage of the log probability leads to the initial probability distribution and the transition model terms falling off from the gradient, as only the policy π_θ is dependent on θ. Therefore the gradient of the log probability function can be found as follows

$$\nabla_\theta \log p_\theta(\tau) = \sum_{k=0}^{H} \nabla_\theta \log \pi_\theta(a_k|s_k) \tag{A.20}$$

This expression can then be used to calculate the gradient by substituting into A.16, to get the gradient estimate $\hat{g}$. To give an estimate of the expected reward $r(\tau)$ over the trajectory, the estimate of the advantage function $\hat{A_k}$, as defined in equation A.11 is used.

$$\hat{g} = \mathbb{E}\{\nabla_\theta \log \pi_\theta(a_k|s_k) \hat{A}_k\} \tag{A.21}$$

The gradient estimate $\hat{g}$ is used in equation A.12 as an approximation for the gradient of the objective function $\nabla_\theta J$. The algorithms which use a gradient estimate to perform the parameter update are known as REINFORCE algorithms [37], or as vanilla policy gradient algorithms, since they are the simplest algorithms which make use of the policy gradient. Most implementations of the algorithm work by differentiating a loss function given by

$$L(\theta) = \mathbb{E}\{\log \pi_\theta(a_k|s_k) \hat{A}_k\} \tag{A.22}$$

The REINFORCE algorithm however suffers from instability during training, low sample efficiency and a lack of robustness [38]. An algorithm based on vanilla gradient algorithms, but with a modified objective function leading to significantly improved performance, is the Proximal Policy Optimization algorithm, which has been discussed in the subsequent section.

A.4. Proximal Policy Optimization

As mentioned above, policy gradient algorithms suffer from very low sample efficiency, and require millions of steps in training to learn an adequate policy. The training is heavily dependent on the step size. A small step size leads to very slow training, while large step sizes could lead to high noise and straying away from the optimal policy. Proximal Policy Optimization (PPO) is a state-of-the-art policy gradient algorithm that improves on the reliability and sample efficiency by using a clipped objective function, which ensures that the updated policy after each step does not stray away regrettably far from the existing policy. The clipped objective function used by the PPO algorithm is as follows

$$L^{CLIP}(\theta) = \mathbb{E}\{\text{min}(r_k(\theta)\hat{A}_k, \text{clip}(r_k(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_k)\} \tag{A.23}$$

where ϵ is a hyperparameter, and $r_k(\theta)$ is the probability ratio $r_k(\theta) = \frac{\pi_{\theta_{h+1}}}{\pi_{\theta_h}}$.

It has been shown that PPO outperforms other algorithms which can deal with continuous state and action spaces, such as Trust Region Policy Optimization (TRPO), Cross-Entropy Methos (CEM), REINFORCE, Advantage Actor Critic (A2C) and Actor-Critic with Experience Replay (ACER) on several MuJoCo environments [38]. A significant advantage of PPO is that the clipped objective function is easy to implement, as opposed to the Kullback–Leibler divergence constrained objective in TRPO, and off-policy corrections and replay buffers required for ACER. PPO has also been applied successfully to automated driving tasks such as vehicle control [39], behavioural decision making [40] and autonomous emergency steering [41]. Due to these reasons, the PPO algorithm was chosen for the motion planning task in this thesis.

B

Custom Environment Definition

The Deep Reinforcement Learning Agent collects data for training by solely interacting with the environment and generating state, action, and reward information. For the purpose of obtaining an agent which performs well in the actual scenario in which it is planned to be implemented, it is of the utmost importance to construct an environment which is representative of the real world problem. Once the agent has been trained, it needs to be evaluated, either in the real world or in a virtual environment. The environment used for evaluation can be the same as or different from the training environment.

In this thesis, a simple environment with a point mass model was used for training to generate cubic spline based motion plans which can be generalised to different types of vehicles, and which keeps training times relatively low. For evaluation, a high-fidelity IPG CarMaker environment was used. This appendix is divided into two sections, with each section detailing the training and evaluation environments respectively.

B.1. Training Environment

The training environment consists of the state space S, the action space A, the state transition function P and the reward function R, which have been previously defined in section A.1. The environment used in this thesis consisted of one-step training episodes, and so the state transition function always resulted in a terminal state, regardless of the state and action taken. The explanation of the reward calculation requires an in-depth review of the literature on motion sickness, and therefore has been explained in appendix C devoted to it. This section goes into the road profile generation, which defines the state space, and the calculation of the motion profile itself, which defines the action space of the agent.

B.1.1. Road Profile Generation

The road profile generated was of a fixed length L for the duration of training. To enable the agent to learn how to navigate corners, the road profile was divided into a number of sectors n_s. Each sector was a circular section with a constant curvature value $\kappa_i, i \in 0, 1, \ldots, s-1$, with a length l_i.

The curvature of each section was sampled from a uniform distribution

$$\kappa_i \sim \mathcal{U}_{[\kappa_{min}, \kappa_{max}]} \tag{B.1}$$

where $\kappa_{min} = -0.1$ and $\kappa_{max} = 0.1$. These values were chosen based on the minimum curvature values encountered in the designated route on which the human driving baseline performance was established, which has been explained more in section B.2.

The lengths of each section needed to be generated such that the cumulative length came out to L, while still providing a uniform distribution over the possible road profile sample space. This was done by first generating a temporary array **t** of $(n_s - 1)$ elements, sampled from a uniform distribution between 0 and 1.

$$\mathbf{t} \sim \mathcal{U}^{n_s}_{[0,1]} \tag{B.2}$$

The array **t** was appended with the elements 0 and 1, and then sorted in an increasing order. The element wise difference of the array was calculated to obtain the sector lengths as fractions f_i of the total length .

$$f_i = t_{i+1} - t_i \quad i \in 0, 1, \ldots, s-1 \tag{B.3}$$

The array **f** of fractions was then multiplied by the total length L of the road profile to obtain uniformly distributed road length partitions l_i. In case of requirement of imposition of a minimum sector length l_{min}, the sector lengths l_i were calculated as

$$l_i = f_i(l - l_{min}) + l_{min} \tag{B.4}$$

This procedure of generating road profile leads to a random profile with uniformly sampled sector lengths l_i and curvatures κ_i. The detailed proof of uniformity of the sampled road lengths can be found in the paper by Smith and Tromble [42]. Together, the curvature and length partitions along with the lateral position and velocity at current time k comprise the state of the environment, given by the array $[\kappa_i, l_i, y_k, v_k]$. It should be noted that all values are normalized to lie between the interval [-1,1].

B.1.2. Motion Profile Calculation

The motion trajectory as calculated by the DRL agent was in the form of a sequence of waypoints defined with respect to the road centerline, along with target velocities at the respective waypoints. The planned motion trajectory can then be used as a reference for a low level motion controller to track.

The size of the action space A of the DRL agent needs to be kept in check to maintain training times and memory requirements in reasonable limits. For instance, predicting the waypoints and velocities for a horizon of 100 m, would require an action space of 200 considering a resolution of 1 m. In addition, DRL output would struggle with ensuring continuity and smoothness of the motion profile. As a workaround to these problems, the motion profile was planned as cubic splines, with separate splines for the waypoints and the vehicle velocities. Cubic splines consist of piece-wise third order polynomials P_i, passing through a set of control points or knot vectors. Cubic splines guarantee C^2 continuity, which ensures continuous acceleration profiles.

It is vital to note here that the use of splines for the motion profile will limit the choice of trajectories of the agent to only the control points, and therefore it can exhibit only limited control over how the vehicle behaves between the control points. The vehicle motion profile here would be dictated by the shapes taken by the splines. This would theoretically lead to a sub-optimal solution as compared to planning for each individual waypoint with a finer resolution, assuming the agent learns the optimal policy in both cases. However, this is a trade-off which was necessary, considering the limited computational power and time available for training the agent.

The motion profile is defined in terms of two splines with k knot vectors each, for lateral deviation y with respect to road center line and velocity v. The distance travelled along the road center line is taken as the independent variable of the polynomials P_i. A normalised variable $u \in [0, 1]$ was taken as the distance parameter, varying from 0 at the beginning of the polynomial section to 1 at the end. For the lateral position, the polynomial is given as

$$P_i(u) = y_i(u) = a_{y,i}u^3 + b_{y,i}u^2 + c_{y,i}u + d_{y,i} \quad i = 0, 1, 2, \ldots k-1 \tag{B.5}$$

The coefficients of the polynomials $a_{y,i}$, $b_{y,i}$, $c_{y,i}$, $d_{y,i}$ were calculated based on the following conditions

- The spline should pass through the knot vectors, therefore the ends of each piece-wise polynomial are constrained to intersect at the control points

$$P_i(0) = y_i, \quad P_i(1) = y_{i+1} \qquad i = 0, 1, 2, \ldots k-1 \tag{B.6}$$

$$a_{y,i} = y_i, \quad a_{y,i} + b_{y,i} + c_{y,i} + d_{y,i} = y_{i+1} \qquad i = 0, 1, 2, \ldots k-1 \tag{B.7}$$

- The spline should be first order continuous, so the first derivative at the end of each polynomial section is equal to the first derivative at the beginning of the subsequent spline.

$$P'_{i+1}(0) = P'_i(1) \qquad i = 0, 1, 2, \ldots k-2 \tag{B.8}$$

$$b_{y,i+1} = b_{y,i} + 2c_{y,i} + 3d_{y,i} \qquad i = 0, 1, 2, \ldots k-2 \tag{B.9}$$

- The second derivative of the spline is also continuous

$$P''_{i+1}(0) = P''_i(1) \qquad i = 0, 1, 2, \dots k-2 \tag{B.10}$$
$$c_{y,i+1} = c_{y,i} + 3d_{y,i} \qquad i = 0, 1, 2, \dots k-2 \tag{B.11}$$

- For the purpose of this work, the boundary conditions imposed on the spline were zero first derivative at the start and end of the motion. This was done to ensure that the vehicle is in a cruising state before and after navigating the road profile, with a constant velocity and heading along the direction of the road.

$$P'_0(0) = 0, \quad P'_{k-1}(1) = 0 \tag{B.12}$$
$$b_{y,0} = 0, \quad b_{y,k-1} + 2c_{y,k-1} + 3d_{y,k-1} = 0 \tag{B.13}$$

All the above equations were collected in the form of a linear system, and solved to find the desired coefficients of the spline.

$$\begin{bmatrix}
1 & 0 & & & & & & & & & & \dots & & 0 \\
0 & 1 & 0 & & & & & & & & & \dots & & 0 \\
1 & 1 & 1 & 1 & 0 & & & & & & & \dots & & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & & & & & & \dots & & 0 \\
0 & -1 & -2 & -3 & 0 & 1 & 0 & & & & & \dots & & 0 \\
0 & 0 & -1 & -3 & 0 & 0 & 1 & 0 & & & & \dots & & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & & & & \dots & & 0 \\
\vdots & & & & & & & & & & & & & \vdots \\
0 & \dots & & & & & & & & & & 1 & 0 & 0 & 0 \\
0 & \dots & & & & & & & -1 & -2 & -3 & 0 & 1 & 0 & 0 \\
0 & \dots & & & & & & & & -1 & -3 & 0 & 0 & 1 & 0 \\
0 & \dots & & & & & & & & & & 0 & 1 & 2 & 3 \\
0 & \dots & & & & & & & & & & 1 & 1 & 1 & 1
\end{bmatrix}
\begin{bmatrix} a_{y,0} \\ b_{y,0} \\ c_{y,0} \\ d_{y,0} \\ a_{y,1} \\ b_{y,1} \\ c_{y,1} \\ d_{y,1} \\ \vdots \\ a_{y,k-1} \\ b_{y,k-1} \\ c_{y,k-1} \\ d_{y,k-1} \end{bmatrix}
=
\begin{bmatrix} y_0 \\ 0 \\ 0 \\ y_1 \\ y_1 \\ 0 \\ 0 \\ y_2 \\ \vdots \\ y_{k-1} \\ 0 \\ 0 \\ y_k \end{bmatrix} \tag{B.14}$$

B.2. Evaluation Environment

To evaluate the real-world performance of the trained DRL agents, two actual roundabouts in the Netherlands were chosen, and the trajectory was planned for 134 m long sections navigating through each roundabout. The roundabouts represent typical daily use-case scenarios encountered by most drivers, which produce accelerations in changing directions, and can provide a good representation of nauseogenicity of different driving styles.

The roundabouts will be referred to as RB1 and RB2, with respective latitude-longitude coordinates (52.06497° E, 4.821414° N) and (52.0674° E, 4.823395° N). Satellite images of both roundabouts along with the respective road profiles as used for evaluation have been shown in figure B.1 and B.2. The road profile information in terms of the lengths and curvatures for RB1 and RB2 have been provided in tables B.1 and B.2 respectively.

Roundabout 1						
Lengths (m)	15.00	22.30	12.82	50.92	14.32	18.64
Curvature (m^{-1})	0.00	-1/54.00	-1/13.04	1/14.52	-1/17.52	0.00

Table B.1: Lengths l_i and curvatures κ_i of the road profile representing roundabout 1

Roundabout 2					
Lengths (m)	17.22	18.46	34.14	20.36	43.82
Curvature (m^{-1})	0.00	-1/17.20	1/19.71	-1/22.09	0.00

Table B.2: Lengths and curvatures of the road profile representing roundabout 2

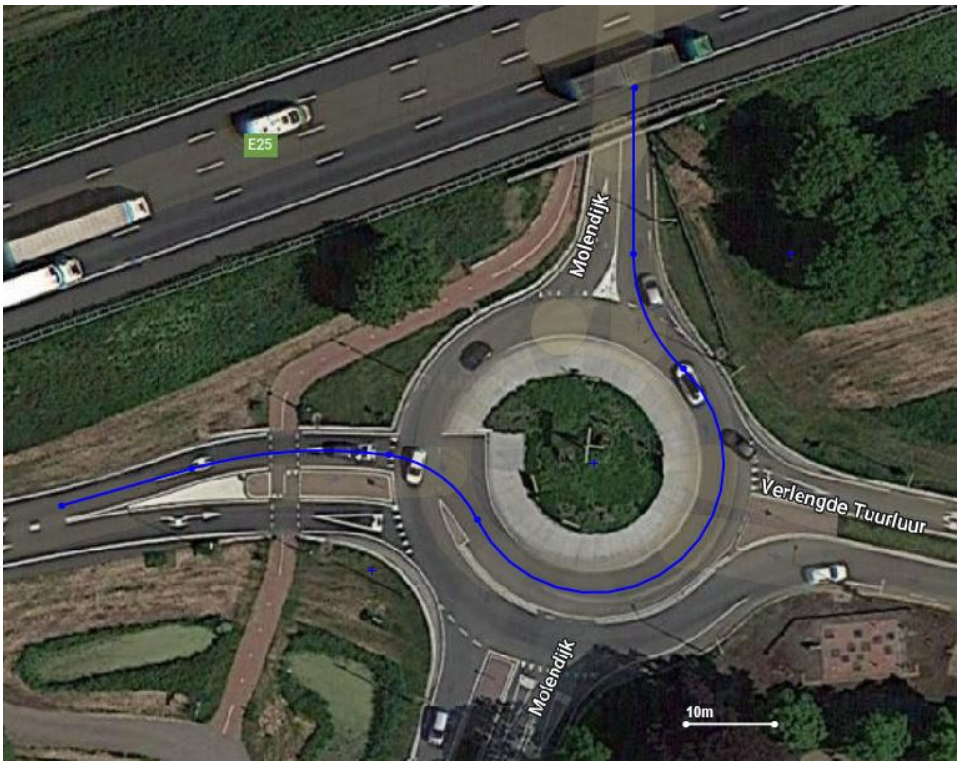

Figure B.1: Satellite view of RB1, along with the road profile used for DRL agent evaluation shown in blue. The road profile length is 134 m, with 6 curvature changes. The scale has been shown at the bottom. Map image courtesy google maps

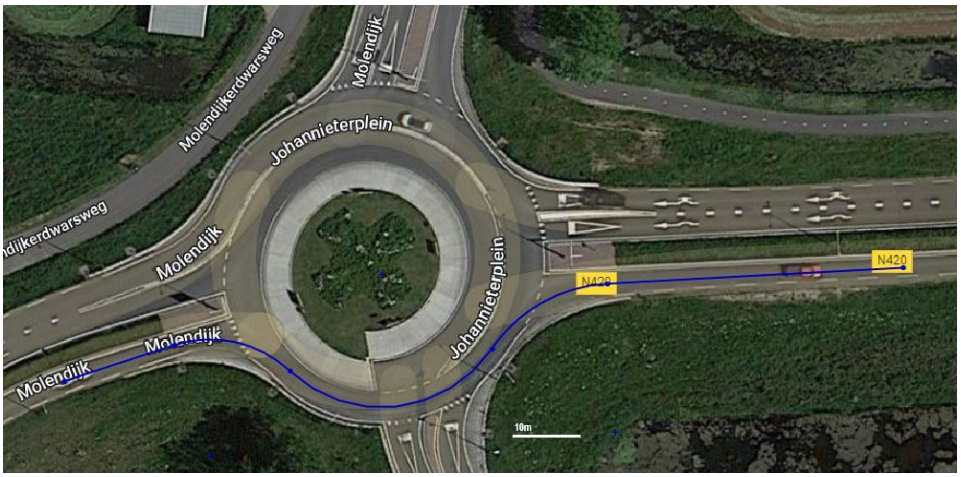

Figure B.2: Satellite view of RB2, along with the road profile used for DRL agent evaluation shown in blue. The road profile length is 134 m, with 5 curvature changes. The scale has been shown at the bottom. Map image courtesy google maps

The road profiles were reconstructed in IPG CarMaker, and the demo Lexus NX300 vehicle model available in IPG was used for simulation. The model was chosen since it was of same weight class, with comparable dimensions and kerb weight as the vehicle used for on-road human driver evaluation, the Hyundai Tucson. An electric powertrain without gearbox was chosen to eliminate the impact of gearshifts on the trajectory following, which is representative of the actual vehicle which was equipped with an automatic gearbox. The initial velocity for each roundabout section was taken equal to the mean value as obtained from the human trials, which came out to 10.40 m/s and 10.46 m/s respectively for RB1 and RB2. To track the reference trajectories generated by the motion planners, a simple Stanley controller was used, which has already been described in the journal paper in Chapter 2. All simulations were carried out with a step size of 5e-3 s, with output step size of 0.01 s, to match the sampling frequency of the sensors used for experimental measurements.

C

Discomfort Evaluation

This appendix lays out the process of evaluating the passenger discomfort to define an appropriate reward function for the DRL agent with the aim of minimizing motion sickness. The relationship between accelerations experienced from the planned motion trajectory and the incidence of motion sickness in passengers needs to be understood in order to design a reward function which can effectively capture the sickening characteristics of the motion plan.

C.1. Motion Sickness Dose Value

Lawther and Griffin in their work explored the relationship between the incidence of motion sickness and the magnitude, frequency and duration of experienced accelerations on board a seafaring vessel [43]. To incorporate the effect of both duration t and magnitude a of imposed accelerations on sickness levels in volunteers, they proposed the calculation of a dose value of the form $a^m t^n$ to quantify the nauseogenicity of the motion regime. As the magnitude of accelerations was found to have a linear relationship with the MSI values, a value of $m = 1$ was assumed. Both $n = 1/2$ and $n = 1/4$ were found to have similar levels of correlation between dose and MSI values. They proposed to use the root of squared integral acceleration values as a dose value, in order to maintain the linear relationship between acceleration and sickness values.

In literature, the most widely used form of the dose value is the Motion Sickness Dose Value, defined as follows

$$MSDV = \sqrt{\int_0^T a_{wf}^2 dt} \tag{C.1}$$

where a_{wf} are the frequency weighted accelerations, using weighting filters which have been discussed in detail in the subsequent section. To define the discomfort term D in this work, only the squared integral term was used, which would accentuate the difference between desirable and undesirable acceleration frequencies due to the shape of the square function. This would further incentivise the minimization of the undesired frequencies. It also allowed to easily sum over the discomfort term over consecutive steps while calculating the trajectory reward. Since the DRL agent was trained in an environment with discrete time, the discomfort was calculated as a sum of the squared accelerations

$$D = \sum_{k=1}^{N-1} (a_{xf,k}^2 + a_{yf,k}^2) \Delta T_k \tag{C.2}$$

ΔT_k is the travel time between stations k and $k+1$. The accelerations $a_{xf,k}$ and $a_{yf,k}$ are the frequency weighted accelerations.

C.2. Frequency Weighting Filters

O'Hanlon and McCauley investigated the frequency dependence of incidence of vomiting in volunteers, and proposed a frequency weighting filter W_f for the design of ride characteristics in land, air and sea

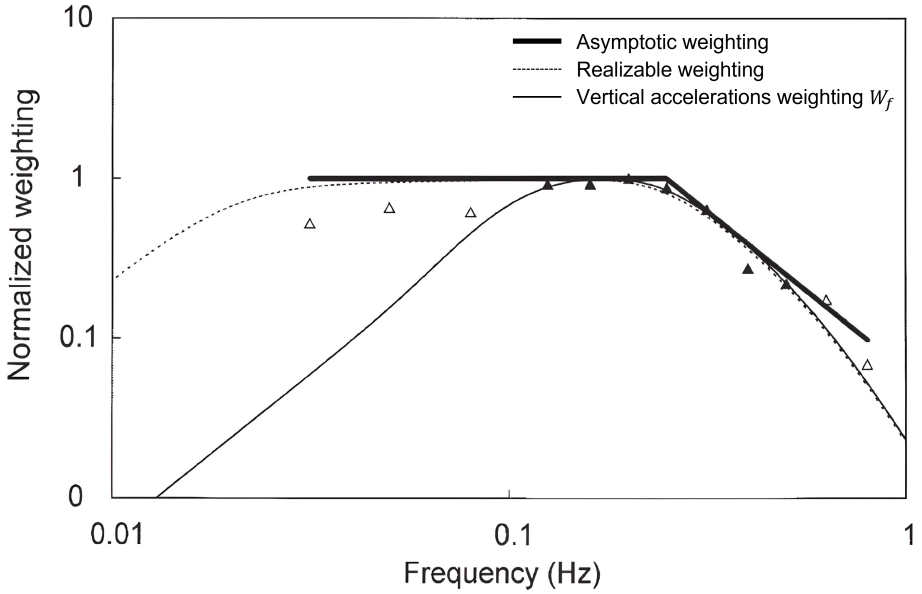

Figure C.1: Asymptotic and realizable frequency weightings for lateral acceleration [11]. The triangles are the measured instances on which the weighting filters have been fit.

vehicles. It was observed that MSI peaked at acceleration frequencies near 0.2 Hz, with tolerance improving appreciably above 0.5 Hz. However, the filter was only based on experiments carried out with vertical oscillations, and cannot be extended directly to horizontal and rotational motions, or any combination of the same. For our purpose of motion planning for road vehicles, it has been shown that lateral accelerations play the most important role, followed by longitudinal oscillations, with no significant contributions from vertical and rotational oscillations [4].

Golding et al found a maximum nauseogenic response of subjects to fore-aft accelerations at a frequency of 0.2 Hz, with the nauseogenicity decreasing with higher and lower frequencies at a slope of 3-4 dB/octave, significantly lower than for vertical oscillations as found by O'Hanlon [44]. The slope decreased further to -4.5 to -5.5 dB/octave for frequencies between 0.35 Hz and 1 Hz [45].

Donohew and Griffin further found that for lateral accelerations, the incidence of mild nausea was independent of the frequency of imposed accelerations between 0.0315 Hz and 0.25 Hz [11]. An asymptotic frequency weighting filter for lateral accelerations was proposed with a slope of 0 dB/octave between 0.0315 Hz to 0.2 Hz, and a slope of -12 dB/octave for frequencies from 0.2 Hz to 0.8 Hz. The asymptotic weighting has been shown in bold in Figure C.1.

Based on the above findings from literature, two separate frequency weighting filters were designed each for lateral and longitudinal accelerations respectively. In the interest of reducing computation time for reward calculation, only first-order high and low pass filters were used to construct the band pass filters. This limited the slope past the cutoff frequencies to 6 dB/octave. This is representative of the slope for longitudinal oscillations as mentioned in [45], but is not steep enough to capture the frequency weighting as described in [11] for lateral accelerations.

The transfer functions of a first order low pass and high pass filter are given as

$$LP(s) = \frac{1}{\tau_1 s + 1} \tag{C.3}$$

$$HP(s) = \frac{s}{\tau_2 s + 1} \tag{C.4}$$

where $\tau_i = 1/(2\pi f_i)$, f_i being the respective cutoff frequency of the filter. For the longitudinal filter the cutoff frequencies were taken as $f_1 = 0.25$ Hz and $f_2 = 0.15$ Hz, while for the lateral filter, the respective

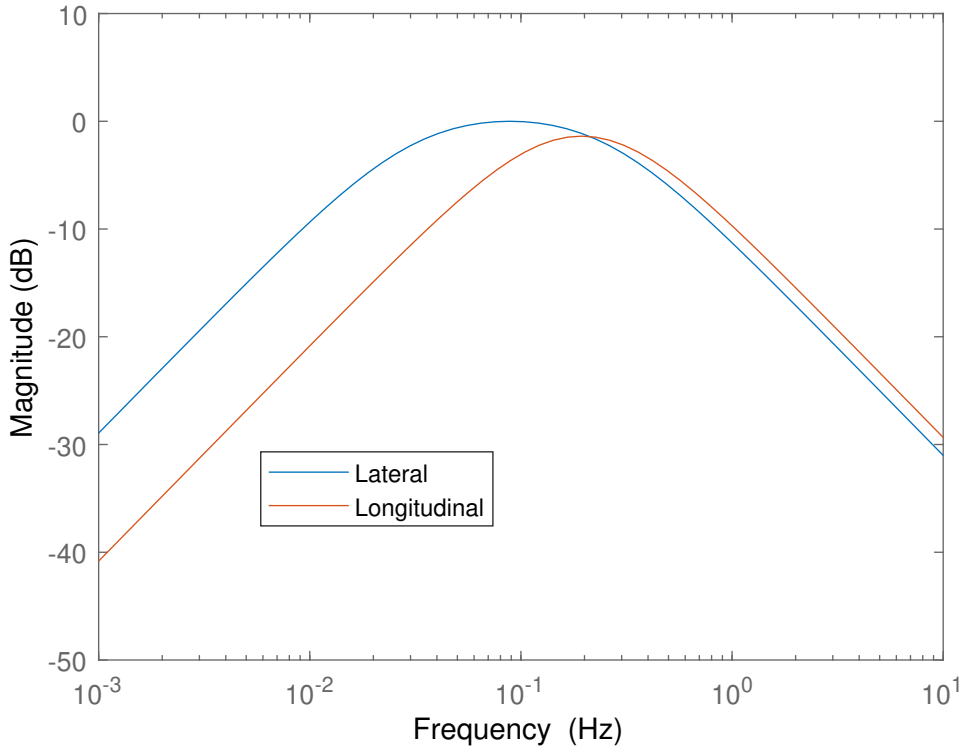

Figure C.2: The normalized band pass filters used for weighting lateral and longitudinal accelerations to calculate the discomfort term

values were $f_1 = 0.25$ Hz and $f_2 = 0.0315$ Hz. The band pass filter was then simply calculated as

$$BP(s) = LP(s).HP(s) = \frac{1}{\tau_1 s + 1}\frac{s}{\tau_2 s + 1} \tag{C.5}$$

The gains of the band pass filters were normalised so that the area under the bode plots were equal for the frequency range 0-1 Hz. This was done to ensure that the magnitudes of frequency weighted accelerations in both directions remain comparable, and accelerations from a particular dimension do not dominate the reward function. The resulting weighting filters have been shown in figure C.2.

C.3. Filter Implementation

In order to apply the frequency weighting filter to the accelerations in discrete time, the transfer function in equation C.5 was first converted to a continuous time state space model

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{C.6}$$

x being an internal state, u being the acceleration a at that time instant, and y being the output of the continuous time band pass filter $BP(s)$. The state space model can then be converted to discrete time by a zero order hold as follows

$$\begin{aligned} A_d &= \mathrm{e}^{A\Delta T} \\ B_d &= \left(\int_{\tau=0}^{T} \mathrm{e}^{A\tau} d\tau\right) B = A^{-1}(A_d - I)B \\ C_d &= C \end{aligned} \tag{C.7}$$

ΔT is the travel time between each station, and is the sampling time used to calculate the discrete time matrices at each step in the trajectory. The matrix exponential is calculated by means of diagonalizing

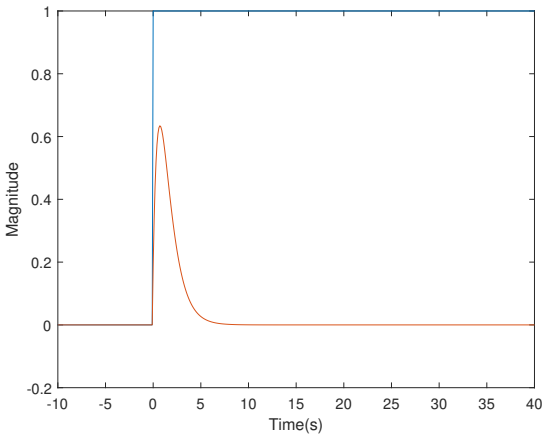

Figure C.3: Step response of longitudinal frequency weighting filter

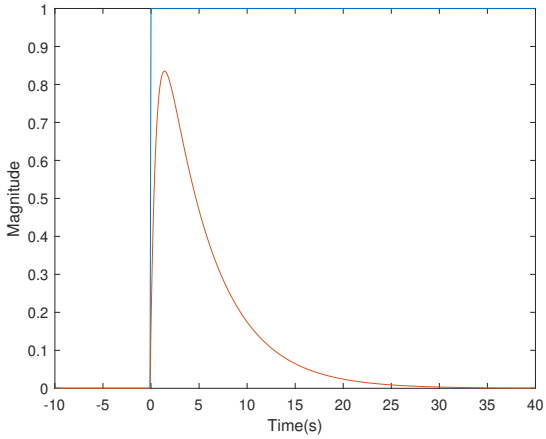

Figure C.4: Step response of lateral frequency weighting filter

the A matrix.

$$\begin{aligned} A &= UDU^{-1} \\ \mathrm{e}^{A\Delta T} &= U\mathrm{e}^{D\Delta T}U^{-1} \end{aligned} \tag{C.8}$$

The matrices U, D, B and C are all calculated offline to minimize computation time.

The step response of the filters have been shown in figures C.3 and C.4. To take into consideration the settling time of the filtered signal, a cooldown period of 10s and 30s was respectively taken for the longitudinal and lateral accelerations, after the end of the planned trajectory. During this period, the accelerations were assumed to be zero, and the output of the filters were penalized.

References

[1] James. Reason and J. J. Brand. *Motion sickness / J. T. Reason, J. J. Brand*. English. Academic Press London ; New York, 1975, xi, 310 p. :

[2] Doreen Huppert, Judy Benson, and Thomas Brandt. “A Historical View of Motion Sickness—A Plague at Sea and on Land, Also with Military Impact”. In: *Frontiers in Neurology* 8 (Apr. 2017).

[3] Eike Schmidt, Ouren Kuiper, Stefan Wolter, Cyriel Diels, and Jelte Bos. “An international survey on the incidence and modulating factors of carsickness”. In: *Transportation Research Part F: Traffic Psychology and Behaviour* 71 (May 2020), pp. 76–87.

[4] Mark Turner and Michael J. Griffin. “Motion sickness in public road transport: The relative importance of motion, vision and individual differences”. In: *British Journal of Psychology* 90.4 (Nov. 1999), pp. 519–530.

[5] Julie Iskander, Attia Mohammed, Khaled Saleh, Darius Nahavandi, Ahmed Abobakr, Shady Mohamed, Houshyar Asadi, Abbas Khosravi, Chee Lim, and Mo Hossny. “From car sickness to autonomous car sickness: A review”. In: *Transportation Research Part F: Traffic Psychology and Behaviour* 62 (Apr. 2019), pp. 716–726.

[6] Bastian Pfleging, Maurice Rang, and Nora Broy. “Investigating user needs for non-driving-related activities during automated driving”. In: Dec. 2016, pp. 91–99.

[7] Arnon Rolnick and Robert Lubow. “Why is the driver rarely motion sick—the Role of controllability in motion sickness”. In: *Ergonomics* 34 (Aug. 1991), pp. 867–79.

[8] Cyriel Diels and Jelte Bos. “Self-driving carsickness”. In: *Applied ergonomics* 53 (Oct. 2015).

[9] J Reason. “Motion Sickness Adaptation: A Neural Mismatch Model”. In: *Journal of the Royal Society of Medicine* 71 (Dec. 1978), pp. 819–29.

[10] James O’Hanlon and Michael McCauley. “Motion Sickness Incidence as a Function of Vertical Sinusoidal Motion”. In: *Aerospace medicine* 45 (May 1974), pp. 366–9.

[11] Barnaby Donohew and Michael Griffin. “Motion sickness: Effect of the frequency of lateral oscillation”. In: *Aviation, space, and environmental medicine* 75 (Sept. 2004), pp. 649–56.

[12] Michael Griffin and Kim Mills. “Effect of frequency and direction of horizontal oscillation on motion sickness”. In: *Aviation, space, and environmental medicine* 73 (July 2002), pp. 537–43.

[13] Larissa Labakhua, Urbano Nunes, Rui Rodrigues, and Fátima Leite. “Smooth Trajectory Planning for Fully Automated Passengers Vehicles: Spline and Clothoid Based Methods and Its Simulation”. In: vol. 15. Jan. 2006, pp. 89–96.

[14] Matthew Mcnaughton, Chris Urmson, John Dolan, and Jin-Woo Lee. “Motion Planning for Autonomous Driving with a Conformal Spatiotemporal Lattice”. In: June 2011, pp. 4889–4895.

[15] Ray Lattarulo, Enrique Martí, Mauricio Marcano, Jose Matute, and Joshué Pérez. “A Speed Planner Approach Based On Bézier Curves Using Vehicle Dynamic Constrains and Passengers Comfort”. In: May 2018, pp. 1–5.

[16] Yanggu Zheng, Barys Shyrokau, and Tamas Keviczky. “3DOP: Comfort-oriented Motion Planning for Automated Vehicles with Active Suspensions”. In: June 2022, pp. 390–395.

[17] Sarah ‘Atifah Saruchi, Mohd Hatta Mohammed Ariff, Hairi Zamzuri, Noor Hafizah Amer, Nurbaiti Wahid, Nurhaffizah Hassan, and Khairil Anwar Abu Kassim. “Novel Motion Sickness Minimization Control via Fuzzy-PID Controller for Autonomous Vehicle”. In: *Applied Sciences* 10.14 (2020).

[18] Mert Sever, Namik Zengin, Ahmet Kirli, and M Selçuk Arslan. “Carsickness-based design and development of a controller for autonomous vehicles to improve the comfort of occupants”. In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 235.1 (2021), pp. 162–176.

[19] Ryosuke Ukita, Yuki Okafuji, and Takahiro Wada. “A Simulation Study on Lane-Change Control of Automated Vehicles to Reduce Motion Sickness Based on a Computational Mode”. In: *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 2020, pp. 1745–1750.

[20] Daofei Li and Jiankan Hu. “Mitigating Motion Sickness in Automated Vehicles With Frequency-Shaping Approach to Motion Planning”. In: *IEEE Robotics and Automation Letters* 6.4 (2021), pp. 7714–7720.

[21] Zaw Htike, Georgios Papaioannou, Efstathios Velenis, and Stefano Longo. “Motion Planning of Self-driving Vehicles for Motion Sickness Minimisation”. In: *2020 European Control Conference (ECC)*. 2020, pp. 1719–1724.

[22] Muhammad Rehan Siddiqi, Sina Milani, Reza N. Jazar, and Hormoz Marzbani. “Ergonomic Path Planning for Autonomous Vehicles-An Investigation on the Effect of Transition Curves on Motion Sickness”. In: *IEEE Transactions on Intelligent Transportation Systems* (2021), pp. 1–12.

[23] Ahmad El Sallab, Mohammed Abdou, Etienne Perot, and Senthil Yogamani. *End-to-End Deep Reinforcement Learning for Lane Keeping Assist*. 2016.

[24] Peter Wolf, Karl Kurzer, Tobias Wingert, Florian Kuhnt, and J. Marius Zollner. “Adaptive Behavior Generation for Autonomous Driving using Deep Reinforcement Learning with Compact Semantic States”. In: *2018 IEEE Intelligent Vehicles Symposium (IV)* (June 2018).

[25] Carl-Johan Hoel, Krister Wolff, and Leo Laine. “Automated Speed and Lane Change Decision Making using Deep Reinforcement Learning”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)* (Nov. 2018).

[26] Pin Wang and Ching-Yao Chan. “Formulation of deep reinforcement learning architecture toward autonomous driving for on-ramp merge”. In: *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)* (Oct. 2017).

[27] Árpád Fehér, Szilárd Aradi, Ferenc Hegedüs, Tamás Bécsi, and Péter Gáspár. “Hybrid DDPG Approach for Vehicle Motion Planning”. In: Jan. 2019, pp. 422–429.

[28] Carl-Johan Hoel, Katherine Driggs-Campbell, Krister Wolff, Leo Laine, and Mykel J. Kochenderfer. “Combining Planning and Deep Reinforcement Learning in Tactical Decision Making for Autonomous Driving”. In: *IEEE Transactions on Intelligent Vehicles* 5.2 (June 2020), pp. 294–305.

[29] Pin Wang and Ching-Yao Chan. *Autonomous Ramp Merge Maneuver Based on Reinforcement Learning with Continuous Action Space*. 2018.

[30] Chris Paxton, Vasumathi Raman, Gregory D. Hager, and Marin Kobilarov. *Combining Neural Networks and Tree Search for Task and Motion Planning in Challenging Environments*. 2017.

[31] Wenpeng Yin, Katharina Kann, Mo Yu, and Hinrich Schütze. “Comparative Study of CNN and RNN for Natural Language Processing”. In: (Feb. 2017).

[32] Ameer Haj-Ali, Nesreen K. Ahmed, Ted Willke, Joseph Gonzalez, Krste Asanovic, and Ion Stoica. *A View on Deep Reinforcement Learning in System Optimization*. 2019.

[33] Szilárd Aradi. “Survey of Deep Reinforcement Learning for Motion Planning of Autonomous Vehicles”. In: *IEEE Transactions on Intelligent Transportation Systems* 23.2 (2022), pp. 740–759.

[34] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. “Human-level control through deep reinforcement learning”. In: *nature* 518.7540 (2015), pp. 529–533.

[35] Hado Van Hasselt, Arthur Guez, and David Silver. “Deep Reinforcement Learning with Double Q-Learning”. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 30 (Sept. 2015).

[36] Ziyu Wang, Nando Freitas, and Marc Lanctot. “Dueling Network Architectures for Deep Reinforcement Learning”. In: (Nov. 2015).

[37] R.J. Williams. “Simple statistical gradient-following algorithms for connectionist reinforcement learning”. In: *Machine Learning* 8 (May 1992), pp. 229–256.

[38] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. “Proximal Policy Optimization Algorithms”. In: (July 2017).

[39] Andreas Folkers, Matthias Rick, and Christof Büskens. *Controlling an Autonomous Vehicle with Deep Reinforcement Learning*. Sept. 2019.

[40] Fei Ye, Xuxin Cheng, Pin Wang, Ching-Yao Chan, and Jiucai Zhang. *Automated Lane Change Strategy using Proximal Policy Optimization-based Deep Reinforcement Learning*. 2020.

[41] Misako Yoshimura, Gakuyo Fujimoto, Abinav Kaushik, Bharat Padi, Matthew Dennison, Ishaan Sood, Kinsuk Sarkar, Abdul Muneer, Amit More, Masamitsu Tsuchiya, Satoru Araki, Anil Hebber, Tijmen Tieleman, and Yuji Yasui. “Autonomous Emergency Steering Using Deep Reinforcement Learning For Advanced Driver Assistance System”. In: Sept. 2020, pp. 1115–1119.

[42] Noah A. Smith and Roy W. Tromble. “Sampling Uniformly from the Unit Simplex”. In: *Tech. rep., Johns Hopkins University* (Aug. 2004).

[43] A Lawther and Michael Griffin. “The motion of a ship at sea and the consequent motion sickness amongst passengers”. In: *Ergonomics* 29 (May 1986), pp. 535–52.

[44] John Golding, AG Mueller, and MA Gresty. “A motion sickness maximum around the 0.2 Hz frequency range of horizontal translational oscillation”. In: *Aviation, space, and environmental medicine* 72 (Apr. 2001), pp. 188–92.

[45] John Golding, M Finch, and J.R.R. Stott. “Frequency effect of 0.35-1.0 Hz horizontal translational oscillation on motion sickness and the somatogravic illusion”. In: *Aviation, space, and environmental medicine* 68 (June 1997), pp. 396–402.