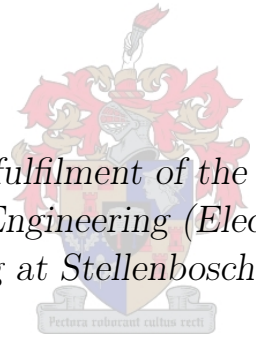
Partial end-to-end reinforcement learning for robustness towards model-mismatch in autonomous racing

by

Andrew Murdoch



Thesis presented in partial fulfilment of the requirements for the degree of Master of Engineering (Electronic) in the Faculty of Engineering at Stellenbosch University

Supervisor: Dr. J.C. Schoeman
Co-supervisor: Dr. H.W. Jordaan

July 2023

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: 2023/07/01 .....

Abstract

The increasing popularity of self-driving cars has given rise to the emerging field of autonomous racing. In this domain, algorithms are tasked with processing sensor data to generate control commands (e.g., steering and throttle) that move a vehicle around a track safely and in the shortest possible time.

This study addresses the significant issue of practical *model-mismatch* in learning-based solutions, particularly in reinforcement learning (RL), for autonomous racing. Model-mismatch occurs when the vehicle dynamics model used for simulation does not accurately represent the real dynamics of the vehicle, leading to a decrease in algorithm performance. This is a common issue encountered when considering real-world deployments.

To address this challenge, we propose a partial end-to-end algorithm which decouples the planning and control tasks. Within this framework, a reinforcement learning (RL) agent generates a trajectory comprising a path and velocity, which is subsequently tracked using a pure pursuit steering controller and a proportional velocity controller, respectively. In contrast, many learning-based algorithms utilise an end-to-end approach, whereby a deep neural network directly maps from sensor data to control commands.

We extensively evaluate the partial end-to-end algorithm in a custom F1tenth simulation, under conditions where model-mismatches in vehicle mass, cornering stiffness coefficient, and road surface friction coefficient are present. In each of these scenarios, the performance of the partial end-to-end agents remained similar under both nominal and model-mismatch conditions, demonstrating an ability to reliably navigate complex tracks without crashing. Thus, by leveraging the robustness of a classical controller, our partial end-to-end driving algorithm exhibits better robustness towards model-mismatches than an end-to-end baseline algorithm.

Uittreksel

Die toenemende gewildheid van selfbesturende motors het aanleiding gegee tot die opkomende veld van outonome wedrenne. In hierdie domein, het algoritmes die taak om sensordata te verwerk om beheeropdragte (bv., stuur en versneller) te genereer wat 'n voertuig veilig en in die kortste moontlike tyd om 'n baan beweeg.

Hierdie studie spreek die beduidende kwessie van praktiese *model-wanverhouding* in leergebaseerde oplossings aan, veral in versterkingsleer (RL), vir outonome wedrenne. Model-wanpassing vind plaas wanneer die voertuigdinamika-model wat vir simulasie gebruik word nie die werklike dinamika van die voertuig akkuraat voorstel nie, wat lei tot 'n afname in algoritme-werkverrigting. Dit is 'n algemene probleem wat teegekom word wanneer werklike implementerings oorweeg word.

Om hierdie uitdaging aan te spreek, stel ons 'n gedeeltelike- 'end-to-end'-algoritme voor wat die beplanning- en beheertake ontkoppel. Binne hierdie raamwerk genereer 'n versterkingsleer (RL) agent 'n trajek wat 'n pad en snelheid bevat, wat vervolgens nagespoor word deur gebruik te maak van 'n suiwer agtervolgstuurbeheerder en 'n proporsionele snelheidsbeheerder, onderskeidelik. Daarteenoor gebruik baie leergebaseerde algoritmes 'n 'end-to-end'-benadering, waardeur 'n diep neurale netwerk direk (DNN) vanaf sensordata karteer om opdragte te beheer.

Ons evalueer die gedeeltelike- 'end-to-end'-algoritme breedvoerig in 'n pasgemaakte 'F1tenth'-simulasie, onder toestande waar model-wanverhoudings in voertuigmassa, draai styfheidskoeffisient en padoppervlakwrywingskoeffisient teenwoordig is. In elk van hierdie scenario's het die werkverrigting van die gedeeltelike- 'end-to-end'-agente dieselfde gebly onder beide nominale en model-wanpastoestande, wat 'n vermoe demonstreer om komplekse spore betroubaar te navigeer sonder om te verongeluk. Deur dus die robuustheid van 'n klassieke kontroleerder te benut, toon ons gedeeltelike- 'end-to-end'- bestuursalgoritme beter robuustheid teenoor model-wanpassings as 'n 'end-to-end'- basislynalgoritme.

Acknowledgements

This thesis appears in its current form due to the assistance and guidance of several people. I would therefore like to offer my sincere thanks to all of them.

I am thankful to God for granting me this opportunity to study. I praise Him for His strength, sustenance, and unwavering faithfulness.

I would like to express my sincere gratitude to my parents, Ross and Jeanne Murdoch. You have been a source of inspiration, and have fostered continual spiritual and emotional growth, as well as provided financial support throughout my studies.

To my supervisors, Dr. J.C. Schoeman and Dr. H.W. Jordaan, I would like to thank you for the guidance that you have provided, as well as the patience and kindness you have shown towards me during my degree. Thank you for the many meetings, comments, corrections, and encouragement.

Friends, thank you for your continual support, prayer, and encouragement throughout my studies.

And to my colleagues at the Electronic Systems Laboratory, thank you for making my studies a pleasant experience.

Contents

List of Figures

List of Algorithms

List of Tables

Nomenclature

Acronyms and abbreviations

LiDAR	light detection and ranging
IMU	inertial measurement unit
DNN	deep neural network
RL	reinforcement learning
MPC	model predictive control
DARPA	Defense Advanced Research Projects Agency
IL	imitation learning
CNN	convolutional neural network
BNN	bayesian neural networks
GTS	Gran Turismo Sport
TORCS	The Open Source Car Simulator
CAPS	conditioning for action policy smoothness
F1	Formula 1
ANN	artificial neural network
ReLU	rectified linear unit
FNN	feedforward neural network
Adam	adaptive moment estimation
MPD	Markov decision process
TD3	twin delay deep deterministic policy gradient
RC	remote controlled
DC	direct current
CoG	centre of gravity

Notation

x	Scalar
$\boldsymbol{x}$	Vector
$\boldsymbol{x}^\intercal$	Transpose of vector $\boldsymbol{x}$

Chapter 1

Racing under model uncertainty

In the previous chapter, we have observed that partial end-to-end agents have an advantage over fully end-to-end agents during both training and testing. However, the comparisons between the two racing algorithms were conducted under conditions that only accounted for uncertainty in the agent's observation by adding noise to the LiDAR scan and vehicle pose. In addition to this, when considering real-world deployment, uncertainties related to the vehicle model also emerge, leading to errors in the vehicle model itself. In this chapter, we compare our partial end-to-end algorithm with the baseline end-to-end algorithm in scenarios where the vehicle model used for evaluation differs from the one utilized during training.

Model mismatch is anticipated to pose a greater challenge in the broader context of road-going autonomous vehicles than racing vehicles, because public roads and cars are not monitored to the same extent as race cars and tracks. However, conducting experiments to assess the impact of model-mismatch on a simulated F1tenth car yields valuable insights applicable to the broader road-going problem. Specifically, we can ascertain which types of model inaccuracies jeopardize vehicle safety and determine the extent to which a given degree of model inaccuracy affects safety and performance. Accordingly, we investigated the types and magnitudes of practical modeling errors that are expected to be found in road-going cars. Our investigations encompass practical modeling errors stemming from three sources:

1. vehicle mass and mass distribution,
2. tire cornering stiffness coefficient, and
3. road surface friction coefficient.

It is important to note that our notion of model mismatch refers to a discrepancy between the vehicle model used during training, and the actual vehicle. Therefore, although online estimation of vehicle model parameters is possible, model mismatch will persist unless the updated vehicle model can be utilized to either retrain the agent online, or replace it with another agent that was trained with a set of vehicle parameters more similar to the real vehicle. These approaches may be prohibitively expensive for complex policies, such as those required by road-going vehicles. Furthermore, while model mismatching is a phenomenon that takes place when deploying agents in the real-world, our analysis is restricted to simulation to prevent unnecessary damage to physical vehicle hardware.

To simulate model-mismatch scenarios, we introduced variations in the vehicle model between the training and evaluation. Algorithm **??** was then employed to evaluate the

agent's performance when racing with the modified vehicle model. This approach is similar to the ones taken by Fuchs et al. [1] and Ghignone et al. [2]. It is important to note that these experiments were conducted on a single agent for each algorithm architecture. The selected agents were representative of the median performance of each framework.

1.1 Adding a dynamic mass

A vehicle's occupants and cargo change its total mass (m), center of gravity relative to the front axle (l_f) and moment of inertia (I_z). The parameters m, I_z and l_f appear in the heading rate and slip angle terms of the dynamic bicycle model described by Equation **??**. However, only l_f appears in the slip angle term for the kinematic bicycle model in Equation **??**. This indicates that a dynamic mass has a greater effect on the vehicle dynamics at higher speeds, where the motion of the vehicle is accurately represented by Equation **??**.

To investigate the effect of a model mismatch in vehicle mass, we simulated the addition of various masses, ranging from 0.3 kg to 1.5 kg, along the longitudinal axis of the vehicle. Subsequently, the performance of both a partial and fully end-to-end agent was evaluated using Algorithm **??**. The percentage of successful laps achieved by the agents is plotted as a function of the position of the masses along the length of the vehicle in Figure 1.1. Furthermore, we chose to perform this experiment on the Porto track, because the difference in performance between the partial and fully end-to-end agents under nominal (i.e., no model mismatch) conditions were minimal on this track.

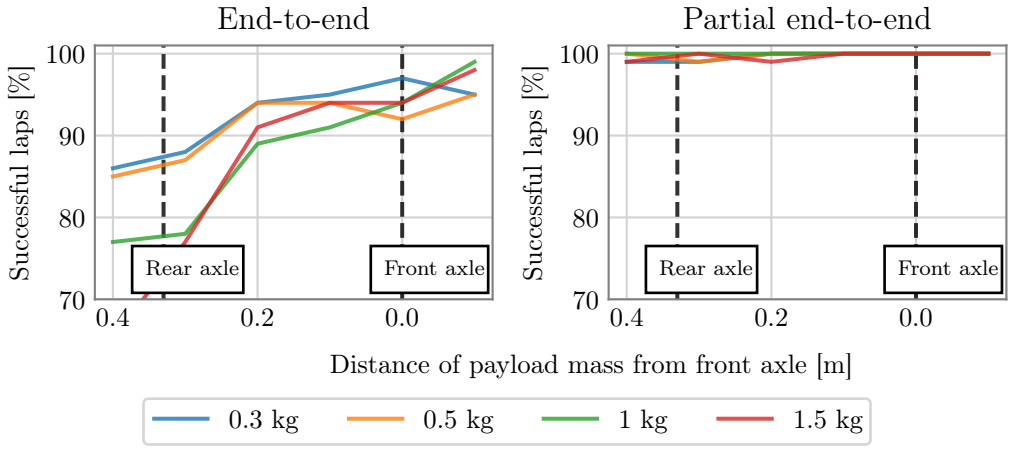


Figure 1.1: The percentage successful laps under evaluation conditions for agents with a masses placed along the longitudinal axis of the vehicle. The front and rear axles are indicated by black dashed lines.

Our findings indicate that the end-to-end agent exhibited sensitivity to an unaccounted-for mass on the vehicle, particularly when the mass was positioned towards the back. This observation is supported by the significant decrease in lap completion rate when the mass is placed closer to the rear axle. Interestingly, the end-to-end agent displayed some resilience towards a mass located at the front of the vehicle. In contrast, the partial end-to-end algorithm demonstrated a higher level of robustness against modeling errors stemming from unaccounted-for masses. Regardless of the position of the mass placement

on the vehicle, the partial end-to-end agent successfully completed a large percentage of it's evaluation laps.

In addition to observing the failure rates of agents with an added mass on the vehicle, we also qualitatively evaluated the behaviour of each agent with a 0.3 kg mass placed directly above the front axle over one lap of the Porto track. Figure 1.2 shows sample trajectories of fully and partial end-to-end agents under nominal conditions, as well as with model-mismatch over one lap.

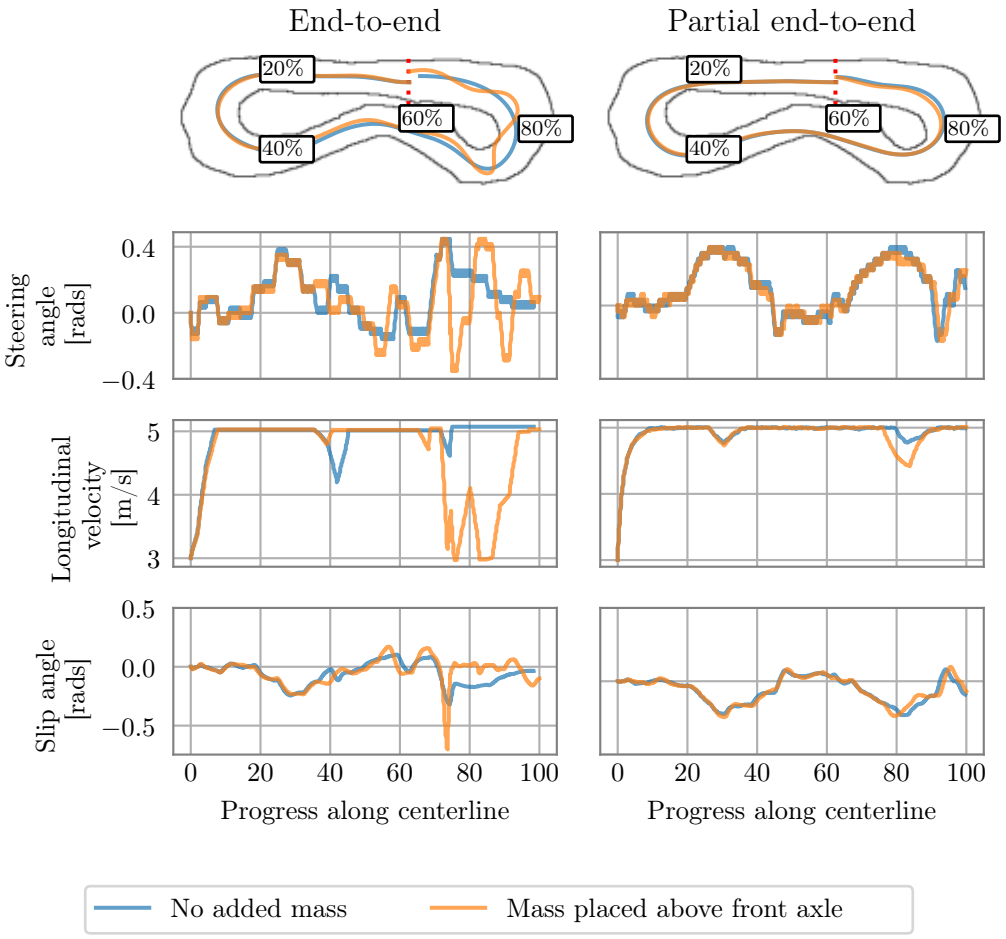


Figure 1.2: The paths, steering angles, longitudinal velocities and slip angles of fully end-to-end agents in the left column, and partial end-to-end agents in the right column, over a single lap of the Porto track. The trajectories of agents with and without the added mass above the front axle are superimposed for comparison.

In Figure 1.2, it is observed that the end-to-end agent, when racing with an added mass, begins to deviate from the nominal trajectory at approximately the halfway point. Initially, the steering angle exhibits slight oscillations. As the agent approaches the corner, these oscillations intensify, accompanied by a significant spike in slip angle, indicating a loss of control. Consequently, the velocity decreases to $v_{\min}$ in an attempt to regain control. However, even after the slip angle returns to the normal range, the agent continues to exhibit a slaloming behavior. Thus, in addition to the fact that end-to-end agents crash on 5% of the evaluation laps, their behaviour is still dangerous on laps that are completed.

In contrast, the partial end-to-end agent did not deviate from the nominal trajectory in any of the evaluation laps. Decoupling path planning from control therefore provides stability and robustness to modeling errors in the total vehicle mass for RL approaches.

1.2 Uncertain cornering stiffness

Another practical model mismatch that vehicles encounter is a discrepancy in the cornering stiffness terms ($C_{S,r}, C_{S,f}$) which characterises their tires. The cornering stiffness is commonly defined as the ratio between the lateral force acting on a tire and its slip angle. Tire construction and dimensions, the type and quality of the tread, and inflation pressure are significant factors when determining cornering stiffness [3]. Once again, the effect of a cornering stiffness model mismatch on the vehicle dynamics is more pronounced at higher speeds, as evident from the absence of tire stiffness terms in the kinematic model applicable at low speeds, and their inclusion in the dynamic model described by Equation **??**.

We conducted an investigation whereby we evaluated the performance of agents after simulating changes in cornering stiffness coefficient for (a) the front tires, (b) the rear tires, then (c) both the front and rear tires together by the same percentage. These changes were once again applied in-between training and evaluation on the Porto track. During these experiments, the tire stiffnesses were varied up to ±20% of the nominal values of 4.72 and 5.45 rads^{-1} for the front and rear tires, respectively. The percentage of successful laps of agents racing with these mismatched cornering stiffness coefficients are shown in Figure 1.3.

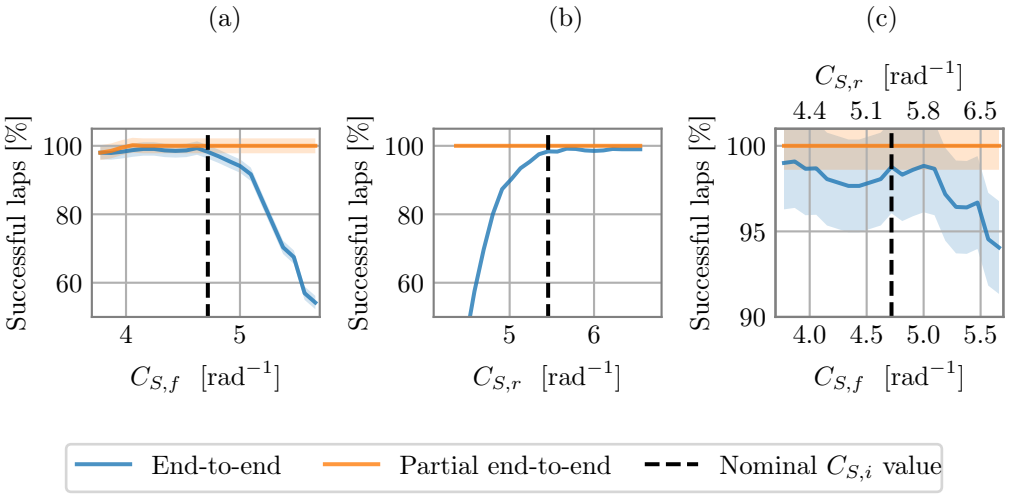


Figure 1.3: Success rates under evaluation conditions of agents with mismatched tire cornering stiffness. Subplot (a) shows the effect of varying only the front tire stiffness, (b) the effect of varying only rear tire stiffness, and (c) the effect of varying both front and rear tire stiffness together.

As is evident from Figure 1.3, the end-to-end agent is sensitive to an increase in the front cornering stiffness coefficient, while also being sensitive to a decrease in the rear cornering stiffness coefficient. In addition, when both the front and rear cornering stiffness coefficients are altered simultaneously, the end-to-end agent tends to experience crashes.

In contrast, the partial end-to-end agent demonstrates resilience to changes in either front or rear tires. Although it does experience failures when both the front and rear cornering stiffness coefficients are decreased together by 20%, the failure rate is comparatively lower than end-to-end agents.

A decrease in rear cornering stiffness coefficient was identified as the worst case scenario for the end-to-end agent. To further investigate this, we compared the trajectories executed by both the partial and fully end-to-end agents on the Porto track, considering a scenario whereby the rear cornering stiffness coefficient was decreased to 4.36 rads^{-1}. Figure 1.4 illustrates the trajectories taken by the agents with and without this model mismatch. In this evaluation, the agents' starting point is at the bottom of the track. When model-mismatch in the rear cornering stiffness coefficient is present, the end-to-end agent exhibits slaloming behavior characterized by significant oscillations in slip angle from the beginning of the lap. In this instance, the end-to-end agent crashes after the second turn.

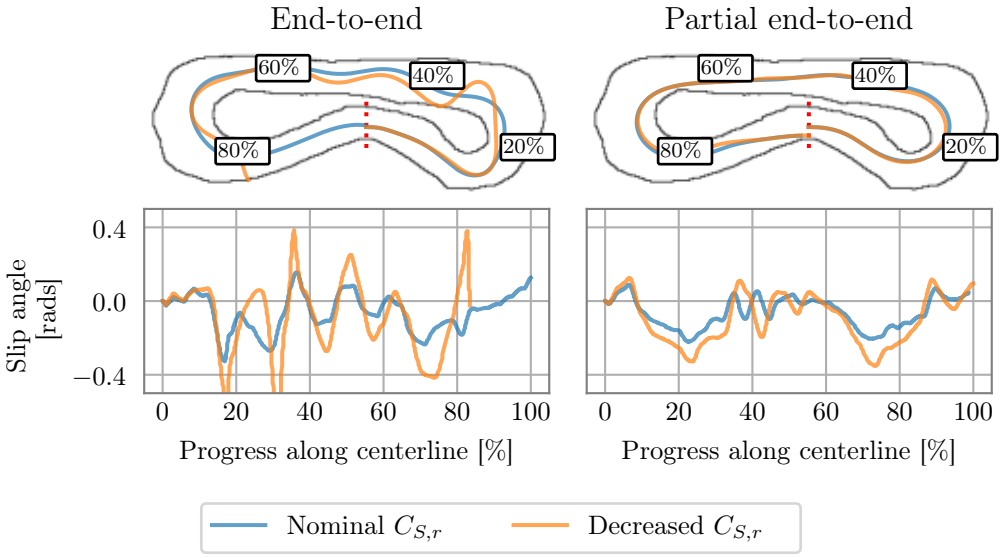


Figure 1.4: Trajectories of agents racing on Porto with and without a decreased rear cornering stiffness coefficient $C_{S,r}$. Trajectories executed by an end-to-end agent are shown in the left column, while trajectories executed by a partial end-to-end agent are shown in the right column. Furthermore, the trajectories of agents racing with and without model mismatch are superimposed for comparison.

In contrast, the trajectories followed by the partial end-to-end agents, with and without the model mismatch, are very similar. Hence, when there is a model mismatch in the cornering stiffness, the partial end-to-end agents outperform the end-to-end agents both quantitatively (in terms of the number of failed laps) and qualitatively (as evident from the analysis of their trajectories). Whereas end-to-end agents tend to slalom when model mismatching occur in the cornering stiffness coefficients, partial end-to-end agents demonstrate more predictable behaviour. Moreover, they did not deviate from the trajectory that they would have executed had there been no model mismatch present.

1.3 Discrepancy in road surface friction

In our final model mismatch investigation, we compared the performance of partial and fully end-to-end agents under conditions where the road surface friction coefficient value used to train the agents is erroneous. Accurately determining the road surface friction coefficient at every point on the road surface is impractical. Moreover, the road surface friction value for a given road surface is influenced significantly by weather conditions. For instance, the minimum friction coefficient values for dry and wet asphalt are 0.7 and 0.4, whereas the minimum road surface friction coefficients for dry and wet gravel are 0.6 and 0.3, respectively [4]. As a result, there is substantial variation of friction values encountered by vehicles on both public roads and racetracks, making a mismatch in friction coefficient highly likely to occur.

For this investigation, we perturbed the road surface friction coefficient in-between training and evaluation. Agents were evaluated with road surface friction values ranging from 0.5 (representing a typical value for wet asphalt) to 1.04 (corresponding to the nominal training value for F1tenth cars given in Table **??**) on all three tracks. Importantly, our simulator assumes a spatially and temporally uniform road friction coefficient throughout a lap. With this assumption, the worst case scenario whereby the entire road surface has a changed friction coefficient is considered.

Figure 1.5 illustrates the percentage of successful evaluation laps for both end-to-end and partial end-to-end agents when facing a mismatch in the road surface friction coefficient across all three tracks. Under nominal conditions, the end-to-end agents achieved success rates of 100%, 79%, and 61% for Porto, Barcelona-Catalunya, and Monaco, respectively. However, when considering the worst case scenario of racing on a wet asphalt surface, the success rates significantly decreased to 46%, 24%, and 26% for these respective tracks.

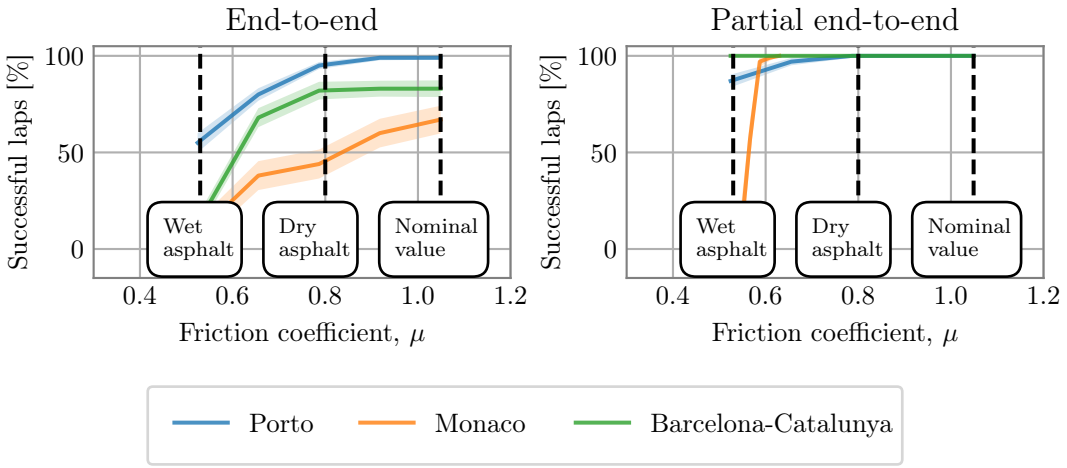


Figure 1.5: Success rates under evaluation conditions of agents racing with decreased road surface friction values. Results for the end-to-end agent racing on all three tracks are shown in the left subplot, while results for the corresponding results for partial end-to-end agents are shown on the right. The values of friction corresponding to the nominal training value, dry and wet asphalt are marked with a black dashed line.

On the other hand, the partial end-to-end agents successfully completed all their evaluation laps under nominal conditions. When a model mismatch in the road surface friction

coefficient was introduced simulating a wet asphalt surface, the decrease in successful laps was less severe compared to the end-to-end agents. The success rate of partial end-to-end agents racing on Porto decreased to 88%, agents racing on Monaco successfully completed 67% of their laps, and those racing on Barcelona-Catalunya completed all of their laps on the wet surface. Thus, partial end-to-end agents consistently outperformed the end-to-end agents regardless of the road surface conditions. Furthermore, the results under model mismatch conditions are consistent across all the tracks that were used in this investigation.

Interestingly, when viewing the locations where agents crashed while racing on a wet asphalt surface on the Monaco circuit, as depicted in Figure 1.6, we see that the majority partial end-to-end agents failed at one of two corners. Conversely, end-to-end agents crashed at various parts of the track while racing with a mismatched friction coefficient.

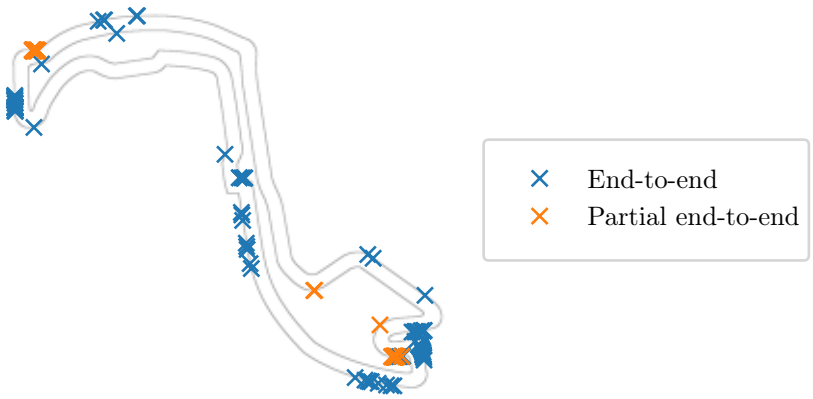


Figure 1.6: Locations where agents crashed while racing on a surface with a friction coefficient of 0.5 (corresponding to wet asphalt).

The fact that partial end-to-end agents fail at only two locations on Circuit de Monaco when racing with a mismatched friction coefficient shows the consistency with which they execute trajectories. To illustruate, an example of trajectories executed by both partial and fully end-to-end agents is shown in Figure 1.7. The figure shows the trajectories executed by agents under nominal conditions, as well as conditions with decreased surface friction. Note that the trajectories executed by the partial end-to-end agent remain similar when mismatches are introduced. The most notable difference between trajectories executed by the partial end-to-end agent on the nominal and slippier surfaces, is that the trajectories executed on the slippier surfaces exhibit reduced curvature, resulting in wider paths being followed by the agents. In contrast, the trajectories by the end-to-end agent are always erratic.

1.4 Summary

In this chapter, we conducted a comprehensive comparison between our partial end-to-end solution and the end-to-end baseline under conditions that mimic real-world deployment

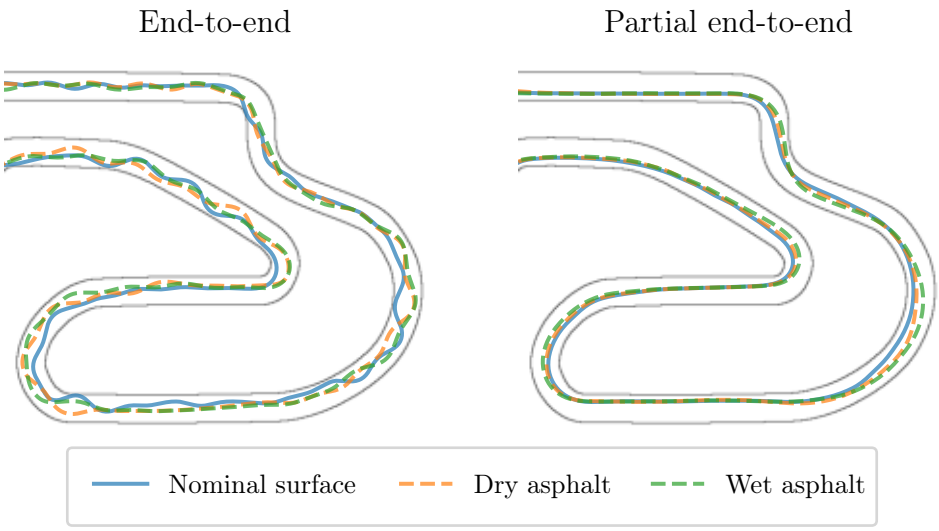


Figure 1.7: Trajectories of agents racing on a section of Barcelona-Catalunya under nominal conditions, as well as with road surface friction values corresponding to dry asphalt, as well as wet asphalt. The trajectories of agents racing with and without model mismatch are superimposed for comparison.

scenarios. Specifically, we introduced model mismatches by using a slightly different vehicle model for training compared to the one used for evaluating the agents. This approach allowed us to simulate the uncertainties often encountered in real-world deployments.

In our investigation, we examined three specific scenarios involving practical model mismatches: errors in (a) the mass, (b) tire cornering stiffness coefficient, and (c) road surface friction coefficient. Across all three scenarios, end-to-end agents performed poorly, as evident by their tendency to crash, as well as to exhibit erratic behavior characterized by slaloming. The erratic behaviour of end-to-end agents highlights their drawbacks in scenarios where model mismatches are present.

In contrast to end-to-end agents, the partial end-to-end agents displayed more predictable behavior. The trajectories of these agents did not deviate significantly from the trajectories observed when racing without any model mismatches. This observation held true across all three model mismatch scenarios considered. These results demonstrate that combining an RL planner with a classic feedback controller is advantageous when considering scenarios in which practical model mismatches are present. Feedback controllers not only offer faster operation but also greater predictability than end-to-end agents, as they are designed to minimize the error between the vehicle and its intended trajectory.

Appendices

Appendix A

Supporting results

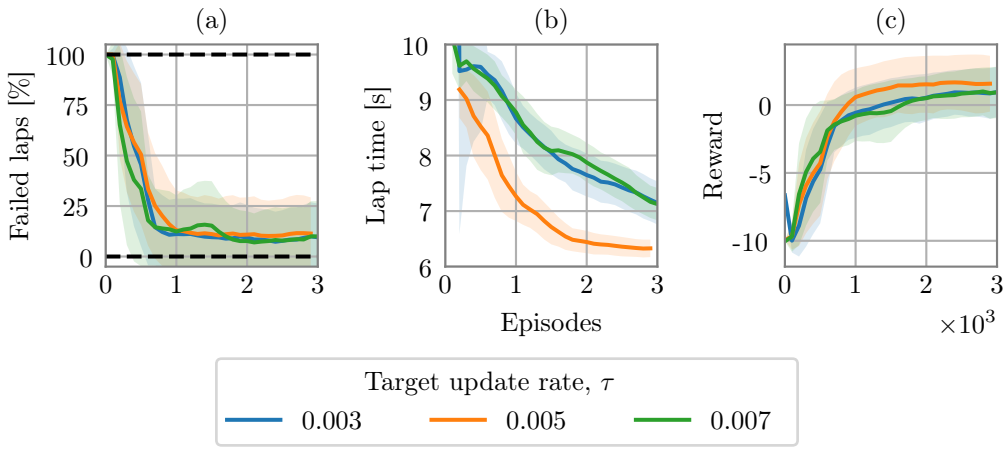


Figure A.1: Learning curves showing (a) the failure rate, i.e percentage of episodes that ended in a crash, (b) the lap time of completed laps, and (b) the episode reward for end-to-end agents with target update rates ranging from 0.003 to 0.007.

Target update rate, τ	**Successful test laps [%]**	**Average test lap time [s]**	**Standard deviation of test lap time [s]**
$3 \cdot 10^{-3}$	99	6.85	1.23
$5 \cdot 10^{-3}$	100	6.07	0.20
$7 \cdot 10^{-3}$	96	6.94	0.74

Table A.1: Evaluation results and training time of end-to-end agents with target update rates ranging from $3 \cdot 10^{-3}$ to $7 \cdot 10^{-3}$.

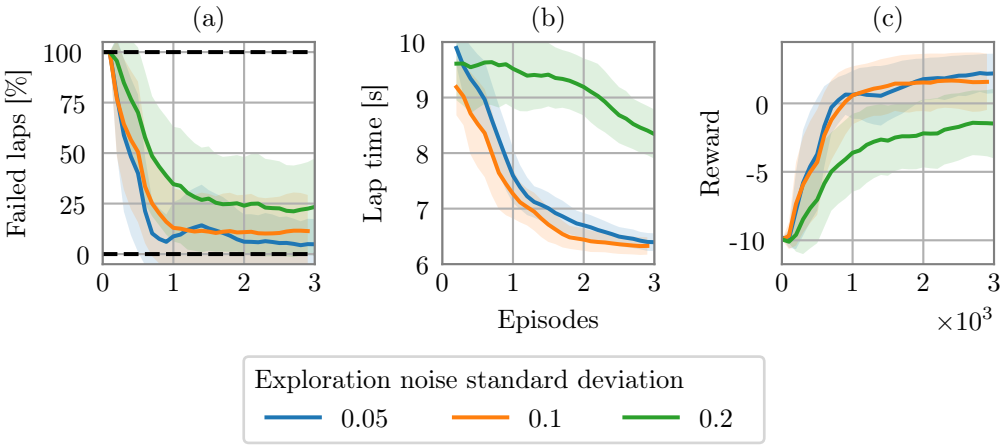


Figure A.2: Learning curves showing (a) the failure rate, i.e percentage of episodes that ended in a crash, (b) the lap time of completed laps, and (b) the episode reward for end-to-end agents with exploration noise standard deviations ranging from 0.05 to 0.2.

Exploration noise standard deviation, σ_{action}	**Successful test laps [%]**	**Average test lap time [s]**	**Standard deviation of test lap time [s]**
0.05	96	6.13	0.46
0.1	100	6.07	0.20
0.2	100	7.27	0.67

Table A.2: Evaluation results and training time of end-to-end agents with exploration noise varying from 0.05 to 0.15.

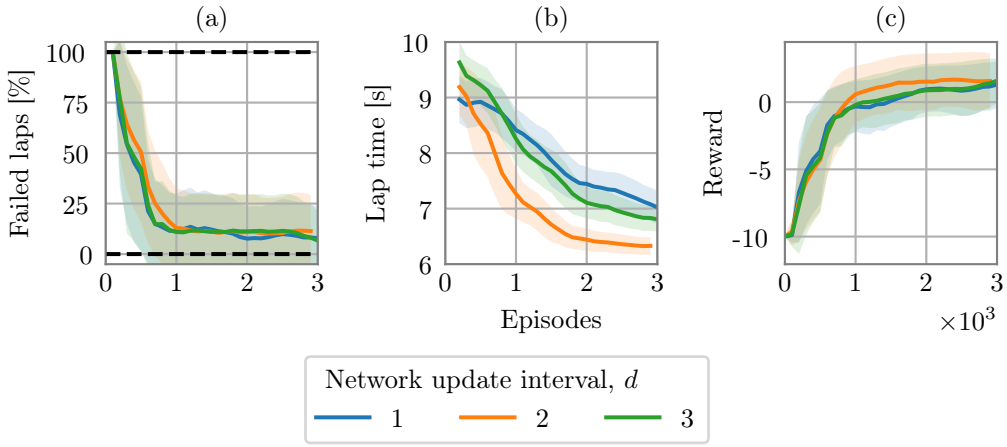


Figure A.3: Learning curves showing (a) the failure rate, i.e percentage of episodes that ended in a crash, (b) the lap time of completed laps, and (b) the episode reward for end-to-end agents with network update intervals d ranging from 1 to 3.

Number of action samples between network updates, d	**Successful test laps [%]**	**Average test lap time [s]**	**Standard deviation of test lap time [s]**
1	99	6.85	1.23
2	100	6.07	0.20
3	96	6.94	0.74

Table A.3: Evaluation results and training time of end-to-end agents with number of action samples between network updates ranging from 1 to 3.

List of References

[1] Fuchs, F., Song, Y., Kaufmann, E., Scaramuzza, D. and Durr, P.: Super-Human Performance in Gran Turismo Sport Using Deep Reinforcement Learning. *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4257–4264, 2021.
Available at: http://doi.org/10.1109/LRA.2021.3064284

[2] Ghignone, E., Baumann, N., Boss, M. and Magno, M.: TC-Driver: Trajectory Conditioned Driving for Robust Autonomous Racing - A Reinforcement Learning Approach. In: *International conference on robotics and automation*. 2022. 2205.09370.
Available at: http://arxiv.org/abs/2205.09370

[3] Vorotic, G., Rakicevic, B. and Mitic, Sasa Stamenkovic, D.: Determination of cornering stiffness through integration of a mathematical model and real vehicle exploitation parameters. *FME Transactions*, vol. 41, pp. 66–71, 2013.
Available at: https://www.mas.bg.ac.rs/_media/istrazivanje/fme/vol41/1/08_gvorotovic.pdf

[4] Novikov, A., Novikov, I. and Shevtsova, A.: Study of the impact of type and condition of the road surface on parameters of signalized intersection. *Transportation Research Procedia*, vol. 36, pp. 548–555, January 2018.
Available at: https://doi.org/10.1016/j.trpro.2018.12.154