Lab Conditions for Research on Explainable Automated Decisions*

Christel Baier[1], Maria Christakis[3], Timo P. Gros[2], David Groß[1], Stefan Gumhold[1], Holger Hermanns[2,4], Jörg Hoffmann[2](✉), and Michaela Klauck[2]

[1] Technische Universität Dresden, Dresden, Germany
[2] Saarland University, Saarland Informatics Campus, Saarbrücken, Germany
[3] Max Planck Institute for Software Systems, Germany
[4] Institute of Intelligent Software, Guangzhou, China

Abstract. Artificial neural networks are being proposed for automated decision making under uncertainty in many visionary contexts, including high-stake tasks such as navigating autonomous cars through dense traffic. Against this background, it is imperative that the decision making entities meet central societal desiderata regarding dependability, perspicuity, explainability, and robustness. Decision making problems under uncertainty are typically captured formally as variations of Markov decision processes (MDPs). This paper discusses a set of natural and easy-to-control abstractions, based on the Racetrack benchmarks and extensions thereof, that altogether connect the autonomous driving challenge to the modelling world of MDPs. This is then used to study the dependability and robustness of NN-based decision entities, which in turn are based on state-of-the-art NN learning techniques. We argue that this approach can be regarded as providing laboratory conditions for a systematic, structured and extensible comparative analysis of NN behavior, of NN learning performance, as well as of NN verification and analysis techniques.

1 Introduction

The field of automated driving – especially in its fully automated form, often referred to as autonomous driving – is considered a grand and worthwhile challenge to tackle. In light of the importance and safety criticality of the application, it is imperative that the core technical components meet societal desiderata regarding dependability, perspicuity, explainability, and trust. The process of automated driving can be broken down into three stages [22]. The first stage deals with machine perception based on the sensor data collected, followed by a stage on

* Authors are listed alphabetically. This work was partially supported by the German Research Foundation (DFG) under grant No. 389792660, as part of TRR 248, see https://perspicuous-computing.science, by the ERC Advanced Investigators Grant 695614 (POWVER), and by the Key-Area Research and Development Program Grant 2018B010107004 of Guangdong Province.

intention recognition, behavior prediction and risk assessment. The last stage is about risk-aware behavioral decisions, the planning of the trajectory to be driven and the effectuation and control of the resulting behavior. Many areas of computer science are in demand here: machine learning, numerics, cyber-physical systems and, last but not least, verification and validation. Across all the three stages, artificial neural networks are being experimented with, and tremendous progress is being reported especially in areas relating to the first and second stage [8,2,4,21,19,24,20].

In formal terms, the interface between the second and third stage can be viewed as a (albeit dauntingly large) Markov decision process (MDP) [18] spanned by a multitude of continuous and discrete dimensions, in which probability annotations reflect the outcomes of risk assessments carried out before. Of course, it is practically infeasible to capture all the precise details of a real vehicle navigating through dense traffic in a single MDP model, and thus it is common practice to instead work on more abstract representations. Typical abstractions are (i) discretization of the continuity of time, space, speed, acceleration and the like, (ii) linearization of non-linearities, and (iii) abstraction from irrelevant details (such as the temperature of the fuel in the vehicle tank) – all that in order to arrive at a model of feasible size. The quality of the abstraction process and the properties of the resulting model are major components in the overall trust we can place on the resulting decision making entity.

In this realm, this paper reports on orchestrated ongoing efforts that aim at systematizing research on (i) the process of abstraction and concretization, and (ii) reproducibility and explainability of decision making entities for automated driving based on neural networks. In a nutshell, we consider the third stage of the automated driving challenge in which a neural network takes the task of risk-aware maneuvering of the vehicle, but in a two-dimensional grid world consisting of blocked and free grid cells, which are observable in full from a bird's-eye view, i. e., without ego-perspective. Furthermore, in a first step, we do not consider moving obstacles, weather or road conditions and resource consumption.

What results after all these abstractions is the problem of navigating a vehicle on a gridded 2D-track from start to goal, as instructed by a neural network and subject to probabilistic disturbances of the vehicle control. This problem family, known as the Racetrack [3] in the AI community is arguably very far away from the true challenges of automated driving, but (i) it provides a common formal ground for basic studies on NN behavioral properties (as we will highlight below), (ii) it is easily scalable, (iii) it is extensible in a natural manner, namely by undoing some of the abstractions listed above (which we are doing already), and (iv) it is made available to the scientific community together with a collection of supporting tools. These four properties are at the core of what we want to advocate with this paper, namely a bottom-up approach to explainability in autonomous driving, providing laboratory conditions for a systematic, structured and extensible analysis. In what follows, we provide a survey of orchestrated ongoing efforts that revolve around the Racetrack case. All infrastructure, documentation, tools, and examples covered in this paper or otherwise related to Racetrack are made available at https://racetrack.perspicuous-computing.science.

2 Racetrack Lab Environment

This section gives a brief overview of the basic model considered and then reviews ongoing work relating to it. The Racetrack evolved from a pen and paper game [10] and is a well known benchmark in AI autonomous decision making contexts [3,17,16,14,5]. In Racetrack, a vehicle needs to drive on a 2D-track (grey) from start cells (green) to goal cells (blue), the track being delimited by walls (red) as depicted in Fig. 1. The vehicle can change acceleration in unit steps in nine directions spanned by the x- and y-dimensions $[-1, 0, 1]$.

The natural abstraction of the autonomous driving challenge in this simplified setting is the task of finding a policy that manages to reach the goal with a probability as high as possible and crashes as rarely as possible. Probabilities enter the picture by imperfect acceleration modelling slippery road conditions.

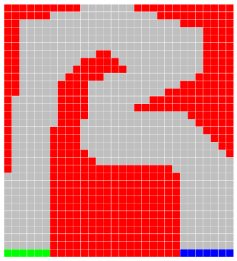

Fig. 1. A Racetrack [3].

We use Racetrack as our lab environment to study various aspects of machine-learnt entities that are supposed to solve the task. These aspects include quantitative evaluations of effectiveness, safety, quality, explainability and verifiability. We will review our work in the remainder of this section. The overarching assumption is that a neural network has been trained by state-of-the-art machine learning techniques for the very purpose of navigating the map as well as possible, and is then put into our lab environment. Beyond that, we also briefly discuss how the lab can be inserted into the machine learning pipeline for the purpose of better learning performance.

2.1 Deep Statistical Model Checking

To enable a deep inspection of the behavior induced by a neural network we developed an evaluation methodology called *Deep Statistical Model Checking* (DSMC) [12]. Concretely, we considered the default Racetrack use case in which the neural network has been trained on the task of reaching the goal with a probability as high as possible while crashing as rarely as possible. After training, the NN represents a policy taking the crucial steering decisions when driving on the map. This policy can be considered as determinizer of the MDP modelling the Racetrack. For the resulting stochastic process, we harvested state-of-the-art statistical model checking [6,23] techniques to study the detailed behavior of the net. More concretely, we treat the NN as a black box to resolve the nondeterminism in the given MDP of the model. The NN gets a description of the current state and returns the action to apply next. The statistical analysis of the resulting Markov chain and thereby of the NN properties

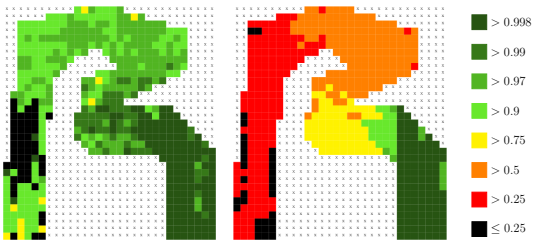

Fig. 2. Effect of expected (left) and more slippery (right) road conditions [12].

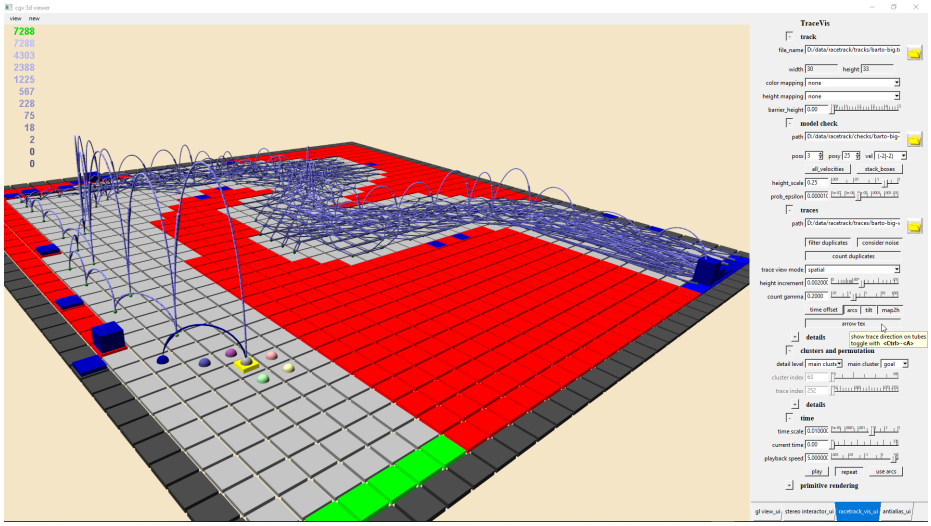

Fig. 3. *TraceVis* in action [11].

gives insights in the quality of the NN, not only for the whole task but also for specific regions of the Racetrack.

An impression is given in Fig. 2, where simple heat maps visualize the chance to safely reach the goal if starting in each of the cells along the track. On the left, the lab model agrees exactly with the model used for training the net, while on the right, it is used in a lab model with a drastically increased probability for the acceleration decision to not take effect (modelling a far more slippery road).

This brief example demonstrates how DSMC enables the inspection of risky driving behavior induced by the NN. Such information can be used to retrain the net in a certain area of the map to improve quality or to see if the net prefers a specific route over an equivalent one. In a nutshell, DSMC provides a scalable verification method of NNs. For small case instances, standard probabilistic model checking can be used, too, for instance to compare the NN behavior with the provably optimal policy, see for more details [12].

2.2 Trajectory Visualization of NN-Induced Behavior

TraceVis [11] is a visualization tool tailored to evaluations in Racetrack-like 2D environments, exploiting advanced 3D visualization technology for data representation and interactive evaluation. In a nutshell, trajectories are mapped to tubes that can be optionally bent to arcs in order to show the discrete nature of stepping, probabilistic information is mapped to a bar chart embedded in the Racetrack or to color, and time can be mapped to a height offset or animation along exploration steps. To reduce visual clutter, segments are aggregated and whole trajectories are clustered by outcome, i.e., final goal or crash position. *TraceVis* offers multiple views for different inspection purposes, including (i) interactive context visualization of the probabilities induced, (ii) visualization of the velocity distribution aggregated from all trajectories, with the possibility to

animate particular aspects as a function of time, and (iii) convenient support for hierarchical navigation through the available clusters of information in an intuitive manner, while still allowing views on individual trajectories. An impression of the UI of the *TraceVis* tool can be found in Fig. 3.

These interactive visualization techniques provide rich support for a detailed inspection of the data space, for the purpose of investigating, for instance, which map positions come with a considerable crash risk, or for what crashes the dominating reason is a bad policy decision taken by the controlling NN, relative to the scenario-intrinsic noise. With these overarching functionalities *TraceVis* offers support for analyzing and verifying neural networks' behavior for quality assurance and learning pipeline assessment in a more detailed and informative way than the raw data and simple heat maps provided in the DSMC work [12].

TraceVis is implemented as a plugin for the CGV-Framework [13], and as such, it is easily extensible to support other dimensions of the autonomous driving challenge.

2.3 Safety Verification for NNs in the Loop

State-of-the-art program analyses are not yet able to effectively verify safety properties of heterogeneous systems, that is, of systems with components implemented using different technologies. This shortcoming becomes especially apparent in programs invoking neural networks – despite their acclaimed role as innovation drivers across many application domains.

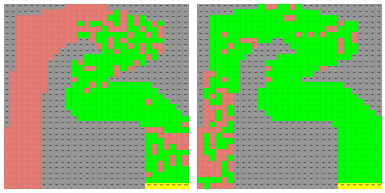

Fig. 4. Effect of training quality on verifiability, for a moderately trained (left) and a well trained (right) NN [7].

We have lately [7] embarked on the verification of system-level safety properties for systems characterized by interacting programs and neural networks. This has been carried out in the lab environment of the Racetrack. The main difference to DSMC is that this work does not consider the net in isolation, but instead takes into consideration the controller program querying the net. Our technique, which is based on abstract interpretation, tightly integrates a program and a neural-network analysis that communicate with each other. For the example case considered, we have for instance studied the dependency between the quality of the NN and the possibility to verify its safety, as illustrated in Fig. 4, where a green cell indicates that it is verifiable that a goal is eventually reached, and red encodes that no property can be verified.

With this work, we address the growing number of heterogeneous software systems and the critical challenge this poses for existing program analyses. Our approach to verifying safety properties of heterogeneous systems symbiotically combines existing program and neural-network analysis techniques. As a result, we are able to effectively prove non-trivial system properties of programs that invoke neural networks.

3 Discussion: Racetrack in the Wild

This paper has discussed several works concentrating on the most basic version of Racetrack. They are supported by a joint software infrastructure, aspects of which are presented in more detail in individual papers. The entirety of the tool infrastructure is available at https://racetrack.perspicuous-computing.science. This web portal presents example tracks to generate Racetrack benchmarks of different sizes and levels of difficulty. Furthermore, it provides demonstrations and explanations how to use the tool infrastructure to

- generate Racetrack versions with different features
- train neural networks on a Racetrack
- perform automated safety verification on a Racetrack
- perform deep statistical model checking on a Racetrack
- explore the resulting behavior with TraceVis.

Beyond this benchmarking infrastructure, what we advocate is a bottom-up approach to autonomous driving (and potentially to other high-stake sequential decision making problems in a similar manner), starting with Racetrack and working upwards to more realism. This endeavor essentially consists in *undoing* the simplifications inherent in the Racetrack benchmark: (i) consideration of resource consumption, (ii) varying road/weather conditions, (iii) moving obstacles, i.e., traffic, (iv) fine discretization, (v) continuous dynamics, (vi) ego-perspective, and (vii) incomplete information. Racetrack can readily be extended with all of these aspects, slowly moving towards a lab environment that encompasses more of the real automated driving challenge. We advocate this as a research road map.

At this point, we are already busy with activities (i) – (iii) of the road map. The basic Racetrack scenario has been extended with different road conditions (tarmac, sand and ice), different engine types that influence maximal speed and acceleration, as well as with tanks of different size, such that the fuel consumption has to be taken into account while driving. Racetrack variants with all these features have been considered in a feature-oriented evaluation study, combined with a hierarchy of different notions of suitability [1]. We are furthermore developing a Lanechange use case, which adds traffic and comes with a switch from a full-observer view, like in Racetrack, to the ego-perspective, where the vehicle has in its view only a certain area to the front, the sides and back. The vehicle here drives on a road with multiple lanes and other traffic participants that move with different speeds in the same direction [15,9]. The aim is to navigate the road effectively, changing lanes to overtake slow traffic, while avoiding accidents. We have developed initial test-case generation methods, adapting fuzzing methods to identify MDP states (traffic situations) that are themselves safe, but on which the neural network policy leads to unsafe behavior.

Overall, we believe that Racetrack, while in itself a toy example, can form the basis of a workable research agenda towards dependability, perspicuity, explainability and robustness of neural networks in autonomous driving. We hope that the infrastructure and research agenda we provide will be useful for AI research in this direction.

References

1. Baier, C., Dubslaff, C., Hermanns, H., Klauck, M., Klüppelholz, S., Köhl, M.A.: Components in probabilistic systems: Suitable by construction. In: Margaria, T., Steffen, B. (eds.) Leveraging Applications of Formal Methods, Verification and Validation: Verification Principles - 9th International Symposium on Leveraging Applications of Formal Methods, ISoLA 2020, Rhodes, Greece, October 20-30, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12476, pp. 240–261. Springer (2020). https://doi.org/10.1007/978-3-030-61362-4_13, https://doi.org/10.1007/978-3-030-61362-4_13
2. Barnaghi, P., Ganz, F., Henson, C., Sheth, A.: Computing perception from sensor data. In: SENSORS, 2012 IEEE. pp. 1–4. IEEE (2012)
3. Barto, A.G., Bradtke, S.J., Singh, S.P.: Learning to act using real-time dynamic programming. Artif. Intell. **72**(1-2), 81–138 (1995). https://doi.org/10.1016/0004-3702(94)00011-O, https://doi.org/10.1016/0004-3702(94)00011-O
4. Berndt, H., Emmert, J., Dietmayer, K.: Continuous driver intention recognition with hidden markov models. In: 11th International IEEE Conference on Intelligent Transportation Systems, ITSC 2008, Beijing, China, 12-15 October 2008. pp. 1189–1194. IEEE (2008). https://doi.org/10.1109/ITSC.2008.4732630, https://doi.org/10.1109/ITSC.2008.4732630
5. Bonet, B., Geffner, H.: Labeled RTDP: improving the convergence of real-time dynamic programming. In: Giunchiglia, E., Muscettola, N., Nau, D.S. (eds.) Proceedings of the Thirteenth International Conference on Automated Planning and Scheduling (ICAPS 2003), June 9-13, 2003, Trento, Italy. pp. 12–21. AAAI (2003), http://www.aaai.org/Library/ICAPS/2003/icaps03-002.php
6. Budde, C.E., D'Argenio, P.R., Hartmanns, A., Sedwards, S.: A statistical model checker for nondeterminism and rare events. In: Beyer, D., Huisman, M. (eds.) Tools and Algorithms for the Construction and Analysis of Systems - 24th International Conference, TACAS 2018, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2018, Thessaloniki, Greece, April 14-20, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10806, pp. 340–358. Springer (2018). https://doi.org/10.1007/978-3-319-89963-3_20, https://doi.org/10.1007/978-3-319-89963-3_20
7. Christakis, M., Eniser, H.F., Hermanns, H., Hoffmann, J., Kothari, Y., Li, J., Navas, J.A., Wüstholz, V.: Automated Safety Verification of Programs Invoking Neural Networks (2020), submitted for publication
8. Dietmayer, K.: Predicting of machine perception for automated driving. In: Autonomous Driving, pp. 407–424. Springer (2016)
9. Faqeh, R., Fetzer, C., Hermanns, H., Hoffmann, J., Klauck, M., Köhl, M.A., Steinmetz, M., Weidenbach, C.: Towards dynamic dependable systems through evidence-based continuous certification. In: Margaria, T., Steffen, B. (eds.) Leveraging Applications of Formal Methods, Verification and Validation: Engineering Principles - 9th International Symposium on Leveraging Applications of Formal Methods, ISoLA 2020, Rhodes, Greece, October 20-30, 2020, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12477, pp. 416–439. Springer (2020). https://doi.org/10.1007/978-3-030-61470-6_25, https://doi.org/10.1007/978-3-030-61470-6_25
10. Gardner, M.: Mathematical games. Scientific American **229**, 118–121 (1973)
11. Gros, T.P., Groß, D., Gumhold, S., Hoffmann, J., Klauck, M., Steinmetz, M.: TraceVis: Towards Visualization for Deep Statistical Model Checking. In: Proceedings of

the 9th International Symposium On Leveraging Applications of Formal Methods, Verification and Validation. From Verification to Explanation. (2020)
12. Gros, T.P., Hermanns, H., Hoffmann, J., Klauck, M., Steinmetz, M.: Deep statistical model checking. In: Gotsman, A., Sokolova, A. (eds.) Formal Techniques for Distributed Objects, Components, and Systems - 40th IFIP WG 6.1 International Conference, FORTE 2020, Held as Part of the 15th International Federated Conference on Distributed Computing Techniques, DisCoTec 2020, Valletta, Malta, June 15-19, 2020, Proceedings. Lecture Notes in Computer Science, vol. 12136, pp. 96–114. Springer (2020). https://doi.org/10.1007/978-3-030-50086-3_6, https://doi.org/10.1007/978-3-030-50086-3_6
13. Gumhold, S.: The computer graphics and visualization framework. https://github.com/sgumhold/cgv, accessed: 18-May-2020
14. McMahan, H.B., Gordon, G.J.: Fast exact planning in markov decision processes. In: Biundo, S., Myers, K.L., Rajan, K. (eds.) Proceedings of the Fifteenth International Conference on Automated Planning and Scheduling (ICAPS 2005), June 5-10 2005, Monterey, California, USA. pp. 151–160. AAAI (2005), http://www.aaai.org/Library/ICAPS/2005/icaps05-016.php
15. Meresht, V.B., De, A., Singla, A., Gomez-Rodriguez, M.: Learning to switch between machines and humans. CoRR **abs/2002.04258** (2020), https://arxiv.org/abs/2002.04258
16. Pineda, L.E., Lu, Y., Zilberstein, S., Goldman, C.V.: Fault-tolerant planning under uncertainty. In: Rossi, F. (ed.) IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013. pp. 2350–2356. IJCAI/AAAI (2013), http://www.aaai.org/ocs/index.php/IJCAI/IJCAI13/paper/view/6819
17. Pineda, L.E., Zilberstein, S.: Planning under uncertainty using reduced models: Revisiting determinization. In: Chien, S.A., Do, M.B., Fern, A., Ruml, W. (eds.) Proceedings of the Twenty-Fourth International Conference on Automated Planning and Scheduling, ICAPS 2014, Portsmouth, New Hampshire, USA, June 21-26, 2014. AAAI (2014), http://www.aaai.org/ocs/index.php/ICAPS/ICAPS14/paper/view/7920
18. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. Wiley Series in Probability and Statistics, Wiley (1994). https://doi.org/10.1002/9780470316887, https://doi.org/10.1002/9780470316887
19. Sadri, F.: Logic-based approaches to intention recognition. In: Handbook of research on ambient intelligence and smart environments: Trends and perspectives, pp. 346–375. IGI Global (2011)
20. Strickland, M., Fainekos, G.E., Amor, H.B.: Deep predictive models for collision risk assessment in autonomous driving. In: 2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018. pp. 1–8. IEEE (2018). https://doi.org/10.1109/ICRA.2018.8461160, https://doi.org/10.1109/ICRA.2018.8461160
21. Tahboub, K.A.: Intelligent human-machine interaction based on dynamic bayesian networks probabilistic intention recognition. J. Intell. Robotic Syst. **45**(1), 31–52 (2006). https://doi.org/10.1007/s10846-005-9018-0, https://doi.org/10.1007/s10846-005-9018-0
22. Wissenschaftsrat: Perspektiven der Informatik in Deutschland (October 2020), https://www.wissenschaftsrat.de/download/2020/8675-20.pdf
23. Younes, H.L.S., Simmons, R.G.: Probabilistic verification of discrete event systems using acceptance sampling. In: Brinksma, E., Larsen, K.G. (eds.) Computer

Aided Verification, 14th International Conference, CAV 2002,Copenhagen, Denmark, July 27-31, 2002, Proceedings. Lecture Notes in Computer Science, vol. 2404, pp. 223–235. Springer (2002). https://doi.org/10.1007/3-540-45657-0_17, https://doi.org/10.1007/3-540-45657-0_17

24. Yu, M., Vasudevan, R., Johnson-Roberson, M.: Risk assessment and planning with bidirectional reachability for autonomous driving. In: 2020 IEEE International Conference on Robotics and Automation, ICRA 2020, Paris, France, May 31 - August 31, 2020. pp. 5363–5369. IEEE (2020). https://doi.org/10.1109/ICRA40945.2020.9197491, https://doi.org/10.1109/ICRA40945.2020.9197491